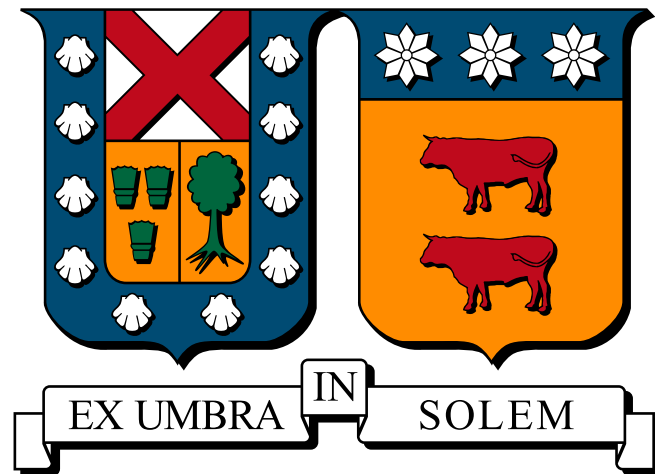


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA



CLASIFICACIÓN DE TEXTOS
MULTI-ETIQUETADOS CON REPRESENTACIÓN
DEPENDIENTE DE LA ETIQUETA

TESIS DOCTORAL

Presentado en satisfacción parcial de los requisitos
para el grado de

DOCTOR EN INGENIERÍA INFORMÁTICA

por

RODRIGO ALFARO A.

Valparaíso, Chile

Marzo de 2021

*Dedicada a mi padre...a mi familia por acompañarme, a
mis amigos por alentarme y a mis profesores por
formarme.*

Índice general

Índice de cuadros	III
Índice de figuras	IX
Resumen	1
Abstract	2
Capítulo 1. Introducción	3
Capítulo 2. Estado del Arte	5
2.1. El Problema de la Clasificación	5
2.2. El Problema de la Clasificación de Textos	7
2.3. Métodos de <i>Ensemble</i>	11
2.4. Clasificadores Lineales	12
2.5. Representación de Textos	15
2.6. Clasificación de Textos Multi-etiquetados	20
Capítulo 3. Marco Metodológico	27
3.1. Conjuntos de Datos	27
3.2. Medidas de desempeño	27
Capítulo 4. Propuesta	31
4.1. Representación Term Frequency-Inverse Document Frequency (<i>tf-idf</i>)	33
4.2. Representación Term Frequency-Relevance Frequency for a Label (<i>tf-rfl</i>)	33
4.3. Representación Multi-Variate Bernoulli Model - Label Dependent (<i>bin-rfl</i>)	34
4.4. Interpretación Probabilística	36
4.5. Interpretación como <i>Ensemble</i>	36
4.6. Interpretación Geométrica	37
4.7. Método de Clasificación	39
Capítulo 5. Resultados Experimentales	41

5.1. Resultados	41
Capítulo 6. Conclusiones y Trabajo Futuro	51
6.1. Conclusiones	51
6.2. Trabajo Futuro	53
Agradecimientos	53
Anexos	55
6.3. Modelo Bernoulli Multivariado	55
6.4. Modelo Multinomial	56
6.5. Tablas con más resultados experimentales	57
Bibliografía	101

Índice de cuadros

2.1. Representación de un Conjunto de Ejemplos Multi-Etiquetados	20
2.2. Transformación Resultante utilizando Relevancia Binaria	21
2.3. Transformación del Problema Usando <i>Label Powerset</i>	22
3.1. Características del Conjunto de Datos Pre-procesados. Cardinalidad (Card) mide el número promedio de etiquetas asociadas con cada documento. Densidad (Dens) se define como la cardinalidad dividida por la cantidad de etiquetas. La Diversidad (Div) representa el porcentaje de conjuntos de etiquetas presentes en el conjunto dividido por el número de conjuntos de etiquetas posibles. El Tamaño del Vocabulario considera el volumen de palabras distintas.	28
4.1. Variables utilizadas para ponderar en un problema multietiqueta dado un término t y 4 categorías	32
5.1. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con SVM en términos de <i>Accuracy</i>	42
5.2. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con ANN en términos de <i>Accuracy</i>	42
5.3. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con SVM en términos de <i>Hamming Loss</i>	43
5.4. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con ANN en términos de <i>Hamming Loss</i>	43

5.5. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con SVM en términos de macro-F_1	44
5.6. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con ANN en términos de macro-F_1	45
6.1. Resultados de Reuters con Binary Relevance- bin-idf	57
6.2. Resultados de Reuters con Binary Relevance- bin-rfl	57
6.3. Resultados de Reuters con Binary Relevance- tf-idf	57
6.4. Resultados de Reuters con Binary Relevance- tf-rfl	58
6.5. Resultados de Reuters con Label Powerset- bin-idf	58
6.6. Resultados de Reuters con Label Powerset- bin-rfl	58
6.7. Resultados de Reuters con Label Powerset- tf-idf	59
6.8. Resultados de Reuters con Label Powerset- tf-rfl	59
6.9. Resultados de Enron con Binary Relevance- bin-idf	59
6.10. Resultados de Enron con Binary Relevance- bin-rfl	60
6.11. Resultados de Enron con Label Powerset- bin-idf	60
6.12. Resultados de Enron con Label Powerset- bin-rfl	60
6.13. Resultados de Slashdot con Binary Relevance- bin-idf	61
6.14. Resultados de Slashdot con Binary Relevance- bin-rfl	61
6.15. Resultados de Slashdot con Label Powerset- bin-idf	61
6.16. Resultados de Slashdot con Label Powerset- bin-rfl	62
6.17. Resultados de Langlog con Binary Relevance- bin-idf	62
6.18. Resultados de Langlog con Binary Relevance- bin-rfl	62
6.19. Resultados de Langlog con Label Powerset- bin-idf	63
6.20. Resultados de Langlog con Label Powerset- bin-rfl	63
6.21. Resultados de Bibtex con con Binary Relevance- bin-idf	63
6.22. Resultados de Bibtex con Binary Relevance- bin-rfl	64
6.23. Resultados de Bibtex con Label Powerset- bin-idf	64
6.24. Resultados de Bibtex con Label Powerset- bin-rfl	64
6.25. Resultados de Genbase con Binary Relevance- bin-idf	65
6.26. Resultados de Genbase con Binary Relevance- bin-rfl	65

6.27.Resultados de Genbase con Label Powerset- bin-idf	65
6.28.Resultados de Genbase con Label Powerset- bin-rfl	66
6.29.Resultados de IMDB con Binary Relevance- bin-idf	66
6.30.Resultados de IMDB con Label Powerset- bin-idf	66
6.31.Resultados de IMDB con Label Powerset- bin-rfl	67
6.32.Resultados de Medical con Binary Relevance- bin-idf	67
6.33.Resultados de Medical con Binary Relevance- bin-rfl	67
6.34.Resultados de Medical con Label Powerset- bin-idf	68
6.35.Resultados de Medical con Label Powerset- bin-rfl	68
6.36.Resultados de Scene con Binary Relevance- bin-idf	68
6.37.Resultados de Scene con Binary Relevance- bin-rfl	69
6.38.Resultados de Scene con Binary Relevance- tf-idf	69
6.39.Resultados de Scene con Binary Relevance- tf-rfl	69
6.40.Resultados de Scene con Label Powerset- bin-idf	70
6.41.Resultados de Scene con Label Powerset- bin-rfl	70
6.42.Resultados de Scene con Label Powerset- tf-idf	70
6.43.Resultados de Scene con Label Powerset- tf-rfl	71
6.44.Resultados de Yeast con Binary Relevance- bin-idf	71
6.45.Resultados de Yeast con Binary Relevance- bin-rfl	71
6.46.Resultados de Yeast con Binary Relevance- tf-idf	72
6.47.Resultados de Yeast con Binary Relevance- tf-rfl	72
6.48.Resultados de Yeast con Label Powerset- bin-idf	72
6.49.Resultados de Yeast con Label Powerset- bin-rfl	73
6.50.Resultados de Yeast con Label Powerset- tf-idf	73
6.51.Resultados de Yeast con Label Powerset- tf-rfl	73
6.52.Resultados de Business con Binary Relevance- bin-idf	74
6.53.Resultados de Business con Binary Relevance- bin-rfl	74
6.54.Resultados de Business con Binary Relevance- tf-idf	74
6.55.Resultados de Business con Binary Relevance- tf-rfl	75
6.56.Resultados de Business con Label Powerset- bin-idf	75
6.57.Resultados de Business con Label Powerset- bin-rfl	75
6.58.Resultados de Business con Label Powerset- tf-idf	76

6.59.Resultados de Business con Label Powerset- tf-rfl	76
6.60.Resultados de Computers con Binary Relevance- bin-idf	76
6.61.Resultados de Computers con Binary Relevance- bin-rfl	77
6.62.Resultados de Computers con Binary Relevance- tf-idf	77
6.63.Resultados de Computers con Binary Relevance- tf-rfl	77
6.64.Resultados de Computers con Label Powerset- bin-idf	78
6.65.Resultados de Computers con Label Powerset- bin-rfl	78
6.66.Resultados de Computers con Label Powerset- tf-idf	78
6.67.Resultados de Computers con Label Powerset- tf-rfl	79
6.68.Resultados de Education con Binary Relevance- bin-idf	79
6.69.Resultados de Education con Binary Relevance- bin-rfl	79
6.70.Resultados de Education con Binary Relevance- tf-idf	80
6.71.Resultados de Education con Binary Relevance- tf-rfl	80
6.72.Resultados de Education con Label Powerset- bin-idf	80
6.73.Resultados de Education con Label Powerset- bin-rfl	81
6.74.Resultados de Education con Label Powerset- tf-idf	81
6.75.Resultados de Education con Label Powerset- tf-rfl	81
6.76.Resultados de Entertainment con Binary Relevance- bin-idf	82
6.77.Resultados de Entertainment con Binary Relevance- bin-rfl	82
6.78.Resultados de Entertainment con Binary Relevance- tf-idf	82
6.79.Resultados de Entertainment con Binary Relevance- tf-rfl	83
6.80.Resultados de Entertainment con Label Powerset- bin-idf	83
6.81.Resultados de Entertainment con Label Powerset- bin-rfl	83
6.82.Resultados de Entertainment con Label Powerset- tf-idf	84
6.83.Resultados de Entertainment con Label Powerset- tf-rfl	84
6.84.Resultados de Health con Binary Relevance- bin-idf	84
6.85.Resultados de Health con Binary Relevance- bin-rfl	85
6.86.Resultados de Health con Binary Relevance- tf-idf	85
6.87.Resultados de Health con Binary Relevance- tf-rfl	85
6.88.Resultados de Health con Label Powerset- bin-idf	86
6.89.Resultados de Health con Label Powerset- bin-rfl	86
6.90.Resultados de Health con Label Powerset- tf-idf	86

6.91.Resultados de Health1 con Label Powerset- tf-rfl	87
6.92.Resultados de Recreation con Binary Relevance- bin-idf	87
6.93.Resultados de Recreation con Binary Relevance- bin-rfl	87
6.94.Resultados de Recreation con Binary Relevance- tf-idf	88
6.95.Resultados de Recreation con Binary Relevance- tf-rfl	88
6.96.Resultados de Recreation con Label Powerset- bin-idf	88
6.97.Resultados de Recreation con Label Powerset- bin-rfl	89
6.98.Resultados de Recreation con Label Powerset- tf-idf	89
6.99.Resultados de Recreation con Label Powerset- tf-rfl	89
6.100.Resultados de Reference con Binary Relevance- bin-idf	90
6.101.Resultados de Reference con Binary Relevance- bin-rfl	90
6.102.Resultados de Reference con Binary Relevance- tf-idf	90
6.103.Resultados de Reference con Binary Relevance- tf-rfl	91
6.104.Resultados de Reference con Label Powerset- bin-idf	91
6.105.Resultados de Reference con Label Powerset- bin-rfl	91
6.106.Resultados de Reference con Label Powerset- tf-idf	92
6.107.Resultados de Reference con Label Powerset- tf-rfl	92
6.108.Resultados de Science con Binary Relevance- bin-idf	92
6.109.Resultados de Science con Binary Relevance- bin-rfl	93
6.110.Resultados de Science con Label Powerset- bin-idf	93
6.111.Resultados de Science con Label Powerset- bin-rfl	93
6.112.Resultados de Social con Binary Relevance- bin-idf	94
6.113.Resultados de Social con Binary Relevance- bin-rfl	94
6.114.Resultados de Social con Binary Relevance- tf-idf	94
6.115.Resultados de Social con Binary Relevance- tf-rfl	95
6.116.Resultados de Social con Label Powerset- bin-idf	95
6.117.Resultados de Social con Label Powerset- bin-rfl	95
6.118.Resultados de Social con Label Powerset- tf-idf	96
6.119.Resultados de Social con Label Powerset- tf-rfl	96
6.120.Resultados de Society con Binary Relevance- bin-idf	96
6.121.Resultados de Society con Binary Relevance- bin-rfl	97
6.122.Resultados de Society con Binary Relevance- tf-idf	97

6.123	Resultados de Society con Binary Relevance- tf-rfl	97
6.124	Resultados de Society con Label Powerset- bin-idf	98
6.125	Resultados de Society con Label Powerset- bin-rfl	98
6.126	Resultados de Society con Label Powerset- tf-idf	98
6.127	Resultados de Society con Label Powerset- tf-rfl	99
6.128	Resultados de tmc2007 con Binary Relevance- bin-idf	99
6.129	Resultados de tmc2007 con Binary Relevance- bin-rfl	99
6.130	Resultados de tmc2007 con Label Powerset- bin-idf	100
6.131	Resultados de tmc2007 con Label Powerset- bin-rfl	100

Índice de figuras

2.1. Arquitectura de una Red Neuronal Perceptrón Multicapa	13
2.2. Hiperplano de decisión basado en el máximo margen	15
4.1. Propuesta como un <i>Ensemble</i>	38
4.2. Interpretación Geométrica	38
5.1. Porcentaje de mejora del desempeño en términos de macro- F_1	46
5.2. Correlación entre las mejoras del desempeño en términos de <i>Hamming Loss</i> y las diferentes características de los set de datos (Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad)	47
5.3. Correlación entre las mejoras del desempeño en términos de <i>Accuracy</i> y las diferentes características de los set de datos (Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad)	48
5.4. Correlación entre las mejoras del desempeño en términos de macro- F_1 y las diferentes características de los set de datos (Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad)	49

Resumen

La asignación de una o más categorías predefinidas a los textos en lenguaje natural, basados en su contenido, es un componente importante y necesario en muchas tareas al interior de las organizaciones. Esta tarea se realiza comúnmente a través de la clasificación automática de textos, esto es, clasificando documentos dentro de un conjunto de categorías predefinidas por medio de un modelo y métodos computacionales. La representación de los textos para propósitos de clasificación automática ha sido tradicionalmente llevada a cabo usando un modelo de espacio vectorial debido a su simplicidad y buen rendimiento. Por otro lado, la clasificación automática de textos por multi-etiquetados ha sido típicamente abordada utilizando métodos de clasificación de etiqueta simple, lo que implica transformar el problema estudiado para aplicar técnicas binarias o adaptar algoritmos binarios para que funcionen con múltiples etiquetas. En esta investigación el objetivo es evaluar una función para la ponderación de las palabras de los textos para modificar la representación de los textos en clasificación multi-etiqueta, usando una combinación de dos enfoques: transformación de problema y adaptación de modelo. Esta función de ponderación y la combinación de enfoques en la clasificación automática fue puesto a prueba con diez diferentes conjuntos de datos textuales utilizados en la literatura especializada y comparado con técnicas alternativas por medio de tres medidas de evaluación. Los resultados presentan mejoras superiores al 10% en el rendimiento de los clasificadores, atribuidas a nuestra propuesta, en todos los casos analizados.

Abstract

The assignment of one or more predefined categories to natural language texts, based on their content, is an important and necessary component in many tasks within organizations. This task is easily carried out through automatic text classification, that is, classifying documents within a set of predefined categories by means of a model and computational methods. The representation of texts for automatic classification purposes has traditionally been carried out using a vector space model due to its simplicity and good performance. On the other hand, the automatic classification of texts by multi-labeling has typically been approached using simple label classification methods, which implies transforming the studied problem to apply binary techniques or adapting binary algorithms to work with multiple labels. In this research the objective is to evaluate a function for the weighting of the words of the texts to modify the representation of the texts in multi-label classification, using a combination of two approaches: problem transformation and model adaptation. This weighting function and the combination of approaches in automatic classification was tested with ten different sets of textual data used in the specialized literature and compared with alternative techniques by means of three evaluation measures. The results show improvements of more than 10 % in the performance of the classifiers, attributed to our proposal, in all the cases analyzed.

Capítulo 1

Introducción

En la era de la explosión de la información, procesar y clasificar grandes cantidades de datos de texto de forma manual requiere mucho tiempo y es un enorme desafío automatizar esta tarea mediante métodos computacionales. Además, la precisión de la clasificación manual de textos puede verse fácilmente influenciada por factores humanos, tales como la fatiga y la experiencia. Lo anterior, exige utilizar métodos de aprendizaje automático para acelerar el procesamiento de clasificación de texto y obtener resultados más confiables y menos subjetivos. Además, esto también puede ayudar a mejorar la eficiencia en la recuperación de información y aliviar el problema de la sobrecarga de información al localizar la información requerida.

El problema de la clasificación automática de textos multietiquetados existe en diferentes dominios. Aún cuando los modelos básicos normalmente plantean la existencia de 2 clases, se han extendido a problemas de más de 2 clases y de multietiqueta, lo que se acerca más a las aplicaciones reales.

Si bien en los últimos años se ha producido un crecimiento en el número de publicaciones basadas en modelos de aprendizaje profundo, no es posible descartar totalmente las técnicas y modelos basados en aprendizaje superficial. Hoy existe un debate entre las aproximaciones de “caja blanca” vs “caja negra”, con ventajas y desventajas para ambos. Del aprendizaje superficial se destaca la factibilidad de trabajar con conjuntos relativamente pequeños de datos y la interpretabilidad. De los modelos de aprendizaje profundo se destaca su buen desempeño y su robustez.

El propósito de esta tesis es evaluar una nueva función de ponderación de términos denominada *relevance frequency for a label* (*rfl*),

presentada inicialmente en [1] para *tf-rfl* y extendida y profundizada en [2] como *bin-rfl*. Se muestra el impacto de la representación considerando diferentes medidas de desempeño para problemas de clasificación de multi-etiqueta. Para ello, se utilizaron dos tipos de clasificadores lineales automáticos usados en este tipo de problemas, estos son: máquina de soporte vectorial lineal (SVM) y redes neuronales artificiales de una capa (ANN). El uso de clasificadores automáticos lineales permite evaluar las mejoras en el rendimiento de los algoritmos, tan solo modificando el espacio de entrada mediante la nueva representación.

La contribución de esta tesis es proponer una nueva representación simple e interpretable, la cual verifica nuestra hipótesis, esto es, que una modificación supervisada a la representación del texto en el espacio de entrada, basada en representaciones binarias o de frecuencias y una ponderación de términos asociada a los ejemplos conocidos según sus etiquetas, puede mejorar significativamente el rendimiento de los clasificadores. Esta nueva propuesta fue validada en diez conjuntos de datos textuales multi-etiquetados, los cuales han sido ampliamente usados en la literatura, con diferentes medidas de desempeño.

Este documento está estructurado de la siguiente forma. En el capítulo 2 se introduce brevemente la clasificación de textos multi-etiqueta y se presenta el estado del arte. En el capítulo 3 se presenta el marco metodológico aplicado. A continuación, en el capítulo 4 se presenta la propuesta. En el capítulo 5 los resultados experimentales donde se compara el desempeño de la propuesta con otros modelos. El último capítulo está dedicado a conclusiones finales, discusión de resultados y trabajos futuros. Finalmente, se presentan en anexos el Modelo Bernoulli Multivariado, el Modelo Multiunomial y tablas con mas resultados experimentales.

Capítulo 2

Estado del Arte

2.1. El Problema de la Clasificación

El objetivo de la clasificación es tomar un vector de entrada \mathbf{x} y asignarlo a una de las K clases discretas C_k donde $k = 1, \dots, K$. En el escenario más común, las clases se toman como disjuntas [3], de modo que a cada entrada se le asigna una y solo una clase. Por tanto, el espacio de entrada se divide en regiones de decisión cuyos límites se denominan límites de decisión o superficies de decisión. En los modelos lineales para la clasificación, las superficies de decisión son funciones lineales del vector de entrada \mathbf{x} y, por lo tanto, están definidas por hiperplanos $(D - 1)$ -dimensionales dentro del espacio de entrada D -dimensional. Se dice que los conjuntos de datos cuyas clases pueden separarse exactamente mediante superficies de decisión lineales son linealmente separables. En el caso de la clasificación, hay varias formas de utilizar valores objetivo para representar etiquetas de clase. Para modelos probabilísticos, lo más conveniente, en el caso de problemas de dos clases, es la representación binaria en la que hay una única variable objetivo $t \in \{0, 1\}$ tal que $t = 1$ representa la clase C_1 y $t = 0$ representa la clase C_2 . Podemos interpretar el valor de t como la probabilidad de que la clase sea C_1 , con los valores de probabilidad tomando solo los valores extremos de 0 y 1. Para $K > 2$ clases, es conveniente usar un esquema de codificación 1-de- K en el que t es un vector de longitud K tal que si la clase es C_j , entonces todos los elementos t_k de t son cero excepto el elemento t_j , que toma el valor 1. Por ejemplo, si tenemos $K = 5$ clases, entonces un patrón de la clase 2 recibiría el vector objetivo

$$(2.1) \quad t = (0, 1, 0, 0, 0).$$

Nuevamente, podemos interpretar el valor de t_k como la probabilidad de que la clase sea C_k . Para los modelos no probabilísticos, las

opciones alternativas de representación de la variable objetivo a veces resultarán convenientes. Según [4], hay tres enfoques distintos para el problema de clasificación. El más simple implica la construcción de una función discriminante que asigna directamente a cada vector \mathbf{x} a una clase específica. Sin embargo, un enfoque más poderoso modela la distribución de probabilidad condicional $P(C_k/\mathbf{x})$ en una etapa de inferencia, y luego utiliza esta distribución para tomar decisiones óptimas. Al separar la inferencia y la decisión, se pueden obtener numerosos beneficios. Hay dos enfoques diferentes para determinar las probabilidades condicionales $P(C_k/\mathbf{x})$. Una técnica consiste en modelarlos directamente, por ejemplo representándolos como modelos paramétricos y luego optimizando los parámetros utilizando un conjunto de entrenamiento. Alternativamente, se puede adoptar un enfoque generativo en el que se modelan las densidades condicionales de clase dadas por $P(\mathbf{x}/C_k)$, junto con las probabilidades *a priori* $P(C_k)$ para las clases, y luego se calculan las probabilidades *a posteriori* requeridas usando el Teorema de Bayes

$$(2.2) \quad P(C_k/\mathbf{x}) = P(\mathbf{x}/C_k)P(C_k)/P(\mathbf{x})$$

En los modelos de regresión lineal, la predicción del modelo $y(\mathbf{x}, \mathbf{w})$ está dada por una función lineal de los parámetros w . En el caso más simple, el modelo también es lineal en las variables de entrada y , por lo tanto, toma la forma $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$, por lo que y es un número real. Para problemas de clasificación, sin embargo, se requiere predecir etiquetas de clase discretas, o más generalmente probabilidades que se encuentran en el rango $(0, 1)$. Para lograr esto, se considera una generalización de este modelo en la que se transforma la función lineal de w usando una función no lineal $f(\mathbf{x}, \cdot)$ de modo que

$$(2.3) \quad y(\mathbf{x}) = f(\mathbf{w}^T \cdot \mathbf{x} + w_0)$$

En la literatura sobre aprendizaje automático, $f(\mathbf{x}, \cdot)$ se conoce como función de activación. Las superficies de decisión corresponden a $y(\mathbf{x}) = \text{constante}$, de modo que $\mathbf{w}^T \mathbf{x} + w_0 = \text{constante}$ y, por tanto, las superficies de decisión son funciones lineales de \mathbf{x} , incluso si la función $f(\mathbf{x}, \cdot)$ no es lineal. Por esta razón, la clase de modelos descritos por la ecuación se denominan modelos lineales generalizados [5].

2.2. El Problema de la Clasificación de Textos

La clasificación de texto (TC: *Text Classification*), también conocida como categorización de texto, es la actividad de asignar etiquetas a textos en lenguaje natural, a partir de un conjunto de categorías predefinidas. TC se remonta a principios de los años 60, pero solo a principios de los 90 se convirtió en un subcampo importante de la disciplina de los sistemas de información, gracias al amplio campo de aplicaciones y al crecimiento de la potencia de cómputo.

Los datos de texto son diferentes de los datos numéricos, de imagen o de señal. Por lo cual, se requiere utilizar técnicas de Procesamiento de Lenguaje Natural (PLN) para preprocesar textos para entregarle al modelo. Los modelos de aprendizaje superficial generalmente necesitan obtener buenas características de los ejemplos mediante diferentes métodos, para luego clasificarlas con algoritmos clásicos de aprendizaje automático. Por lo tanto, la eficacia del método de clasificación está supeditada, en gran medida, por la extracción de las características del texto. Sin embargo, a diferencia de los modelos superficiales, el aprendizaje profundo integra la extracción de características en el proceso de ajuste del modelo mediante el aprendizaje de un conjunto de transformaciones no lineales que sirven para asignar características directamente a los resultados.

En [6] se plantea que los primeros métodos basados en modelos estadísticos (aprendizaje superficial), tales como Naive Bayes [7], K-Nearest Neighbor [8], Support Vector Machine [9] dominaron hasta el año 2010. Desde la década de 1960 hasta la de 2010, dominaron los modelos de clasificación de texto basados en el aprendizaje superficial. En comparación con los métodos anteriores basados en reglas, este método tiene ventajas obvias en cuanto a precisión y estabilidad. Este enfoque necesita realizar selección de características, lo que puede requerir mucho tiempo y costos. Desde la década de 2010, la clasificación de texto se ha extendido utilizando modelos de aprendizaje profundo. En comparación con los métodos basados en el aprendizaje superficial, los métodos de aprendizaje profundo evitan el diseño de reglas y características por parte de humanos y proporcionan automáticamente representaciones semánticamente significativas para el análisis de texto. No obstante,

la clasificación de textos basada en aprendizaje profundo requiere de muchos datos y tiene una alta complejidad computacional.

Los modelos de clasificación de aprendizaje superficial requieren de una buena representación del texto, basada en la extracción de características. Entonces, una vez que la representación del texto tiene características bien diseñadas, el clasificador puede converger rápidamente y entregar un buen desempeño.

Por su parte, hasta finales de los años 80, el enfoque más popular de la CT, al menos en la comunidad “operativa” (es decir, aplicaciones del mundo real), era el de ingeniería del conocimiento (EC), que consistía en definir manualmente un conjunto de reglas que codificaban el conocimiento experto sobre cómo clasificar documentos en las categorías dadas. A partir de los años 90, este enfoque se ha ido perdiendo (especialmente en la comunidad de investigadores) a favor del paradigma del aprendizaje automático (ML), según el cual un proceso inductivo general construye automáticamente un clasificador de texto, mediante algún tipo de aprendizaje a partir de un conjunto de documentos preclasificados, reconociendo las características de las categorías de interés. Las ventajas de este enfoque son una precisión comparable a la lograda por los expertos humanos, y un ahorro considerable en términos de mano de obra experta, ya que no se necesita la intervención de expertos en el dominio para la construcción del clasificador o para su portabilidad a un conjunto diferente de categorías. Este trabajo se concentra en el enfoque *Machine Learning* de la Clasificación de Textos.

Desde la década de 1960 hasta la de 2010, dominaron los modelos de clasificación de texto basados en el aprendizaje superficial. En [6] se plantea que los primeros métodos basados en modelos estadísticos (aprendizaje superficial), tales como Naive Bayes [7], K-Nearest Neighbor [8], Support Vector Machine [10] dominaron hasta el año 2010. En comparación con los métodos anteriores basados en reglas, este método tiene ventajas obvias en cuanto a precisión y estabilidad. Este enfoque necesita realizar selección de características, lo que puede requerir mucho tiempo y costos. Desde la década de 2010, la clasificación de texto se ha extendido utilizando modelos de aprendizaje profundo. En comparación con los métodos basados en el aprendizaje superficial, los métodos

de aprendizaje profundo evitan el diseño de reglas y características por parte de humanos, y proporcionan automáticamente representaciones semánticamente significativas para el análisis de texto. No obstante, la clasificación de textos basada en aprendizaje profundo requiere de muchos datos y tiene una alta complejidad computacional.

Los modelos de clasificación de aprendizaje superficial requieren de una buena representación del texto, basada en la extracción de características. Entonces, una vez que la representación del texto tiene características bien diseñadas, el clasificador puede converger rápidamente y entregar un buen desempeño.

TC se está aplicando ahora en muchos contextos, que van desde la indexación de documentos basada en un vocabulario controlado, hasta el filtrado de documentos, la generación automatizada de metadatos, la desambiguación del sentido de las palabras, la población de catálogos jerárquicos de recursos web y, en general, cualquier aplicación que requiera organización de documentos o selección y despacho adaptativo de documentos.

Los métodos basados en aprendizaje superficial aprenden de los datos, donde la representación de las características del texto es muy importante para el alcanzar un buen desempeño de clasificación. Sin embargo, requiere del desarrollo de la selección de características. Antes de entrenar al clasificador, necesitamos recopilar conocimientos o experiencia para extraer características del texto original. Los métodos de aprendizaje superficial entrenan al clasificador inicial basándose en varias características textuales extraídas del texto sin procesar. Según [?] para conjuntos de datos pequeños, los modelos de aprendizaje superficial generalmente presentan un mejor rendimiento que los modelos de aprendizaje profundo bajo la limitación de la complejidad computacional. Por lo tanto, algunos investigadores han estudiado el diseño de modelos superficiales para dominios específicos con menos datos. Por su parte, los modelos de aprendizaje profundo pueden aprender representaciones de características directamente de la entrada sin demasiadas intervenciones manuales y conocimientos previos. Sin embargo, la metodología de aprendizaje profundo es un método basado en datos, que

generalmente necesita una gran cantidad de datos para lograr un alto rendimiento. Normalmente se requieren muchos más datos que en los algoritmos tradicionales de aprendizaje automático, lo que significa que esta técnica no se puede aplicar a tareas de clasificación en conjuntos de datos pequeños [11]. Además, la enorme cantidad de datos necesarios para los algoritmos de clasificación de aprendizaje profundo exacerba aún más la complejidad computacional durante el paso de entrenamiento [12]. Finalmente, la falta de interpretabilidad que estos modelos tienen, no los hace comparable con modelos superficiales para explicar por qué y cómo funciona bien.

Ambos, los modelos superficiales y profundos, pueden lograr un buen rendimiento en la mayoría de las tareas de clasificación de texto. No obstante, la interpretación de los modelos profundos aún es un desafío.

En los últimos años, los investigadores han diseñado muchos modelos para mejorar la precisión de los modelos de clasificación de texto. Sin embargo, cuando hay algunos ejemplos complejos en los conjuntos de datos, el rendimiento del modelo disminuye significativamente. En consecuencia, la relación entre eficacia y eficiencia en los modelos, y cómo mejorar la robustez de los modelos es un desafío y un foco de las investigaciones actuales.

Los modelos de aprendizaje profundo tienen ventajas únicas en la extracción de características y minería semántica y han logrado excelentes resultados en las tareas de clasificación de texto. Sin embargo, el aprendizaje profundo es un modelo de caja negra, el proceso de entrenamiento es difícil de reproducir. Asimismo, son desafíos pendientes la comprensión de la semántica implícita y la interpretación de los resultados.

El modelo superficial mejora el rendimiento de la clasificación de texto principalmente al mejorar el esquema de extracción de características y el diseño del clasificador. Por el contrario, el modelo de aprendizaje profundo mejora el rendimiento al aprender características implícitas de los datos sin procesar, la estructura del modelo, además

de los datos y conocimiento adicionales que utiliza en las fases previas.

La interpretabilidad de los modelos de aprendizaje profundo, especialmente *Deep Neural Network*, siempre ha sido un factor limitante para los casos de uso que requieren explicaciones de las características involucradas en el modelado y tal es el caso de muchos modelos prescriptivos aplicados al área de salud [13] o en automatizaciones que pueden afectar la libertad de las personas [14], [15]. Los pesos en una red neuronal son una medida de qué tan fuerte es cada conexión entre cada neurona para encontrar el espacio de características importantes. Por ello, los algoritmos complejos como el aprendizaje profundo son difíciles de entender.

Tal como se plantea en [16], el aprendizaje profundo es una de las técnicas más poderosas en inteligencia artificial (IA), y muchos investigadores y científicos se enfocan en modelos de aprendizaje profundo para mejorar la robustez y el poder computacional de esta herramienta. Sin embargo, los modelos de aprendizaje profundo también tienen algunas desventajas y limitaciones cuando se aplican a tareas de clasificación. Uno de los principales problemas de este modelo es que el aprendizaje profundo no facilita una comprensión teórica completa del aprendizaje. Una desventaja bien conocida de los métodos de aprendizaje profundo es su naturaleza de “caja negra”. Es decir, el método por el cual los métodos de aprendizaje profundo generan la salida convolucionada no es fácilmente comprensible.

2.3. Métodos de *Ensemble*

El aprendizaje de *Ensemble* estudia el uso de la integración de un conjunto de modelos para construir un predictor que mejore el rendimiento sobre un modelo único más complejo.

Según [17], hay dos supuestos subyacentes a los algoritmos de conjuntos de modelos: el primero es que se puede construir un predictor con mayor precisión combinando modelos, en lugar de usar un modelo de mayor complejidad. Así, el promedio ponderado de resultados de una colección de modelos podría mejorar el rendimiento de la predicción de un conjunto de datos como una combinación lineal de ellos,

podría generar un sesgo más bajo que cualquiera de los predictores individuales (incluso pudiendo elegir el mejor predictor individual).

La segunda premisa sugiere que el rendimiento de un conjunto es mejor que un predictor basado en un modelo. Por lo general, las medidas de pérdida, por ejemplo, la pérdida cuadrática, no se ven afectadas por un cambio en el modelo (si sus predicciones permanecen inalterables). Las predicciones individuales se comparan entre sí, ya que solo dependen de la predicción y la observación. En cambio, los enfoques basados en modelos podrían violar el principio de preeminencia al confundir la construcción del predictor con su desempeño.

El uso de múltiples copias de un clasificador individual no produce una mejora real en la generalización del conjunto y, por lo tanto, parece intuitivo que cuando se combinan predictores es necesario que tengan algún grado de heterogeneidad o diversidad. La diversidad de un conjunto puede fomentarse implícitamente modificando el conjunto de datos para el entrenamiento del clasificador, modificando la arquitectura de los predictores o los parámetros de aprendizaje. Mientras que los métodos explícitos para construir conjuntos, utilizando una métrica de diversidad que depende de los otros miembros del conjunto, aunque no se garantiza necesariamente que la diversidad contribuya a mejorar el rendimiento del conjunto.

Los métodos de *ensemble* pueden diferir en tres etapas diferentes: manipulación de datos, manipulación del clasificador base y en la función que utiliza el comité para generar la salida de consenso.

2.4. Clasificadores Lineales

De modo de analizar el impacto de la modificación de la función de representación de textos, en esta investigación se utilizan 2 clasificadores lineales para realizar la predicción: Máquinas de Soporte Vectorial (SVM) y Redes Neuronales Artificiales (ANN) de una capa oculta.

2.4.1. Redes Neuronales Artificiales Multicapa. La idea de las redes neuronales nace de emular los sistemas biológicos que funcionan como clasificadores naturales de patrones. Una neurona artificial se describe por el producto interno del vector de entradas \mathbf{x} con el vector de pesos \mathbf{w} , seguido por una función de activación no-lineal, tal como la función sigmoidea $sgm(x) = 1/(1 + e^{-x})$, o radio basal, entre otras. La arquitectura típica de las redes neuronales se llama perceptrón multicapa, donde existen conexiones completas entre capas adyacentes. La capa de entrada sólo recibe las entradas, y no realiza ningún procesamiento. La capa oculta realiza un primer procesamiento con neuronas no-lineales, y su salida sirve de entrada a la capa de salida.

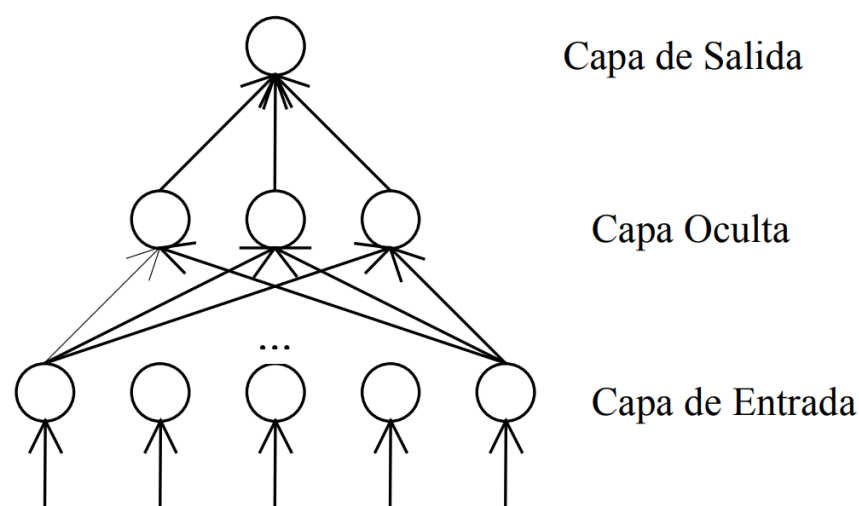


FIGURA 2.1. Arquitectura de una Red Neuronal Perceptrón Multicapa

Cada característica del vector de entradas $\mathbf{x} = (x_1, \dots, x_D)$ alimenta un nodo de la capa de entradas.

A su vez, cada nodo de la capa oculta contiene otra característica, que es una función de la combinación lineal de las entradas.

Finalmente, cada nodo de la capa de salida es una función de la combinación lineal de las salidas de los nodos de la capa oculta.

Combinando estas diversas etapas se obtiene la función de red general que, para las funciones de activación de la unidad de salida sigmoidea, toma la forma

$$(2.4) \quad y_k(\mathbf{x}, \mathbf{w}) = \text{sgm} \left(\sum_{j=1}^M w_{kj}^{(2)} \cdot \text{sgm} \left(\sum_{i=1}^D w_{ji}^{(1)} \cdot x_i + w_{j0}^{(1)} \right) + w_{k0}^{(2)} \right)$$

donde el conjunto de todos los parámetros de ponderación y sesgo se ha agrupado en un vector \mathbf{w} . Por lo tanto, el modelo de red neuronal es simplemente una función no lineal de un conjunto de variables de entrada $\{x_i\}$ a un set de variables de salida $\{y_k\}$ controladas por un vector de parámetros \mathbf{w} , que se encuentra minimizando el error de entrenamiento y prueba, sobre los ejemplos conocidos.

2.4.2. Máquinas de Soporte Vectorial. Un conjunto de datos $\mathbf{X} = \{x_1, \dots, x_m\} \subseteq \mathbb{R}^N$ que representa ejemplos de dos clases diferentes, etiquetados como $\mathbf{Y} = \{y_1, \dots, y_m\} \subseteq \{T_1 = 1, T_2 = -1\}$ se dice linealmente separable si es posible encontrar un hiperplano que los separe completamente, de acuerdo a las clases a las que pertenecen. Un hiperplano que separe los ejemplos en sus respectivas clases, se puede parametrizar con \mathbf{w} y b . Una posible implementación de la función (parametrizable) de decisión es $f(\mathbf{x}, \cdot) = \text{sign}(w_0 \cdot x + b)$, donde $\mathbf{x}, \mathbf{w} \in \mathbb{R}^N$. Si los datos son linealmente separables, se puede encontrar un par de parámetros \mathbf{w}, b que satisfaga la ecuación. Éstos pueden ser escalados de tal forma que cumplan con $|f(x_p, w, b)| = 1$, donde x_p es el dato de entrenamiento más cercano al plano de decisión. En estos fundamentos se basan las SVM para identificar el hiperplano que maximice la distancia entre los ejemplos más cercanos de ambas clases, con lo que también se maximiza la capacidad de generalización del modelo.

Por tanto, la distancia de un punto \mathbf{x}_n a la superficie de decisión está dada por

$$(2.5) \quad \frac{t_n \cdot y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n \cdot (\mathbf{w}^T \phi(\mathbf{x}_n) + b)}{\|\mathbf{w}\|}.$$

El problema de encontrar la mayor distancia entre los conjuntos es equivalente a identificar el hiperplano separador que maximiza la distancia entre las clases. El margen viene dado por la distancia perpendicular al punto más cercano \mathbf{x}_n del conjunto de datos, y deseamos optimizar los parámetros \mathbf{w} y b para maximizar esta distancia. Por lo tanto, la solución de margen máximo se encuentra resolviendo

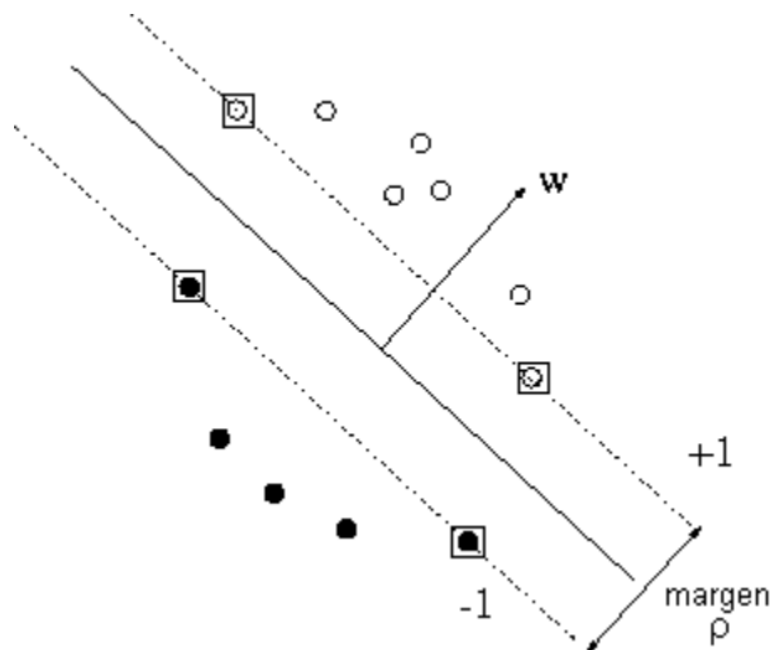


FIGURA 2.2. Hiperplano de decisión basado en el máximo margen

$$(2.6) \quad \arg \max_{\mathbf{w}, b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[(t_n \cdot (\mathbf{w}^T \phi(\mathbf{x}_n) + b)) \right] \right\}$$

Existen diversos métodos de optimización para encontrar el hiperplano único.

2.5. Representación de Textos

El análisis de texto es un campo interdisciplinario relativamente nuevo, basado en las ciencias de la computación y la lingüística. Hoy, la disponibilidad de alta capacidad computacional, los nuevos lenguajes de programación y las múltiples fuentes de datos, han fomentado el desarrollo de nuevos métodos desde las humanidades y las ciencias sociales, en conjunto con los lingüistas computacionales y la ciencia de la computación.

A partir de esta convergencia, surge la clasificación automática de textos como una tecnología que permite asignar etiquetas pre definidas a textos de lenguaje natural, basándose en su contenido. Esta tecnología ha logrado posicionarse como un componente importante de

muchas tareas de las organizaciones y su administración. Su aplicación más importante y común hasta la fecha ha sido en la asignación de categorías de tema a un documento para apoyar el filtrado de textos, ruteo de correos electrónicos, identificación de lenguaje, evaluación de lecturabilidad [18], análisis de polaridad [19], perfilamiento de autores [20], análisis de opiniones [21], entre otros. La clasificación automática de textos juega un papel relevante en una gran variedad de tareas de administración de información las cuales son más flexibles, dinámicas y personalizadas, entre otras, ordenamiento de correos electrónicos u organización jerárquica de carpetas en tiempo real, identificación de tema para el soporte de operaciones, revisar búsquedas o navegación estructurada, encontrar documentos que correspondan a intereses de largo plazo o intereses más dinámicos basados en tareas.

El rendimiento de un sistema de razonamiento basado en casos depende en gran parte de la representación del problema. La misma tarea puede ser fácil o difícil, dependiendo en la forma en la cual está descrita [22]. La representación explícita de información relevante tiende a aumentar el desempeño de la máquina de aprendizaje. De esta manera, a través de una representación más compleja podrían obtenerse mejores resultados con algoritmos más simples. En esta línea, hoy se plantean mecanismos que permiten seleccionar y ponderar las características que mejoran el desempeño de un clasificador automático en contextos específicos [23]. Así también, quienes plantean el análisis de los elementos visuales del texto como una característica adicional [21].

Tradicionalmente, la clasificación (o categorización) de textos se ha definido como la asignación de un valor Booleano (verdadero o falso) a cada par $\langle d_j, c_i \rangle \in D \times C$, donde D es el dominio de los documentos (corpus) y $C = c_1, \dots, c_{|c|}$ es el conjunto de etiquetas (clases) predeterminadas. Si un documento está categorizado solo bajo una etiqueta, es decir, en alguna partición del conjunto de clases (categorías no sobrepuestas) o bajo múltiples etiquetas a la vez (categorías sobrepuestas), se le llama un ‘problema de una etiqueta’ o un problema multi-etiqueta’ respectivamente [24]. El caso más estudiado para resolver problemas de clasificación de texto es el de ‘una etiqueta’ y el enfoque principal es el llamado de Clasificación Binaria (Binary classification, BC), donde

un documento es clasificado, ya sea a la categoría c_i o a su complemento ($\neg c_i$). Este enfoque puede ser extendido y utilizado para resolver problemas con más categorías.

En el caso particular de los documentos, la representación del texto tiene un alto impacto en la tarea de clasificación [25]. El modelo espacio vectorial es uno de los modelos más usados para recuperación de información, principalmente por su simplicidad conceptual y el atractivo de su metáfora subyacente, de usar una proximidad espacial para proximidad semántica [26]. En el modelo de espacio vectorial (*Vector Space Model*), los contenidos de un documento son representados por un vector de términos $d = w_1, \dots, w_k$, donde k es el tamaño del conjunto de términos w_i (o características). Algunos elementos usados en la representación de un texto son los N -gramas, palabras, frases, lógica de términos y declaraciones o cualquier otra unidad léxica, semántica y/o sintáctica que pueda ser utilizada para representar el contenido del texto.

Independientemente de las características usadas para representar un texto, a partir de la existencia de estas características se determinará a qué clases pertenece el texto. Si se considera sólo la existencia de la característica, o bien, si se considera la frecuencia de aparición de esa característica, se pueden distinguir 2 modelos, que [27] denomina Modelo Bernoulli Multi-variado y Modelo Multinomial, respectivamente.

El Modelo Multinomial especifica que un documento está representado por el conjunto de apariciones de términos del documento. El orden de los términos se pierde, sin embargo, se captura el número de apariciones de cada término en el documento. Si se utiliza un modelo Bayesiano, al calcular la probabilidad de un documento, se multiplica la probabilidad de que aparezcan los términos. Aquí se puede entender que las apariciones de términos individuales son los eventos y el documento como la colección de eventos de términos. La medida más utilizada es el indicador de relevancia $f_{(t,d)}$, el cual se utiliza para representar cuánto contribuye la característica o término t a la semántica del documento d y puede tener valores entre cero y uno ($[0,1]$).

Por su parte, el Modelo Bernoulli Multi-variado especifica que un documento está representado por un vector de atributos binarios que indica qué términos ocurren y qué términos no ocurren en el documento. No se captura el número de veces que aparece un término en un documento. Al igual que en el caso anterior, si se utiliza un modelo Bayesiano, al calcular la probabilidad de un documento, se multiplica la probabilidad de todos los valores del atributo, incluida la probabilidad de no ocurrencia de términos que no aparecen en el documento. Aquí se puede entender que el documento es el evento y la ausencia o presencia de términos como atributos del evento. Esto describe una distribución basada en un modelo de eventos de Bernoulli multi-variado. En este caso, se utiliza el indicador $bin_{(t,d)}$, el cual toma el valor 1 cuando el término t existe al menos una vez en el documento d , es decir, puede tener valor cero o uno ($\{0,1\}$). El factor basado en el Modelo Bernoulli Multi-variado es llamado representación Binaria o modelo Booleano. Muchos problemas, ya sea por su naturaleza o por las medidas que se pueden obtener de ellos, utilizan el modelo de representación basado en el Modelo Bernoulli Multi-variado [27].

Existen diferentes formas de describir las características de un texto para que los diferentes clasificadores de textos puedan trabajar sobre ellos. [28] y [10], por ejemplo, combinan transformaciones con diferentes funciones de kernel en máquinas de soporte vectorial. Por su parte, de acuerdo a [29] se deben tomar dos decisiones importantes al escoger la representación basada en el modelo de espacio vectorial. 1) ¿qué debería constituir un término? ¿debería ser una raíz de palabra, una palabra, un conjunto de palabras o su significado? 2) ¿cómo deberá ser pesado o ponderado ese término? La ponderación podría ser por medio de una función binaria o de la frecuencia inversa en los documentos ($tf - idf$) desarrollada por [30], usando métricas de selección de características como chi-cuadrado (χ^2), ganancia de información (IG), razón o ratio de ganancia (GR), etc. Los métodos de ponderación de términos mejoran la efectividad de la clasificación de textos a través de una apropiada selección de pesos para los términos. Aunque la clasificación de textos se ha estudiado durante varias décadas, los métodos de ponderación de los términos para la clasificación de textos suelen tomarse del campo de la recuperación de información (IR), incluyendo, por ejemplo, el modelo Booleano, $tf - idf$ y sus variantes. En general,

para ponderar los términos en el modelo de espacio vectorial, se puede utilizar la frecuencia de términos o frecuencia de documentos que contienen un término.

Según [31], muchas investigaciones han trabajado la técnica de extracción de características del texto intentando mantener la relación sintáctica y semántica que se pierde cuando sólo se consideran las palabras. Y aunque muchos investigadores abordaron técnicas novedosas para resolver este problema, aún mantienen limitaciones. Una solución simple pero costosa para el problema sintáctico es utilizar la técnica n -gram para la extracción de características.

La forma más básica de extracción de características de palabras ponderadas es la Frecuencia del Término (del inglés, *Term Frequency: TF*), donde cada palabra se asigna a un número correspondiente al número de apariciones de esa palabra en todo el corpus. Los métodos que amplían los resultados de TF generalmente usan la frecuencia de palabras como una ponderación booleana o escalada logarítmicamente. En todos los métodos de ponderación de palabras, cada documento se traduce a un vector (con una longitud igual a la del documento) que contiene la frecuencia de las palabras en ese documento. Aunque este enfoque es intuitivo, está limitado por el hecho de que determinadas palabras que se utilizan comúnmente en el idioma pueden dominar tales representaciones. Por el contrario, las funciones de palabras ponderadas se basan en el recuento de palabras en documentos y se pueden utilizar como un esquema de puntuación simple de representación de palabras. Cada técnica presenta limitaciones únicas. Las palabras ponderadas calculan la similitud del documento directamente desde el espacio de recuento de palabras, lo que aumenta el tiempo de cálculo para vocabularios grandes. Si bien los recuentos de palabras únicas proporcionan evidencia independiente de similitud, no tienen en cuenta las similitudes semánticas entre palabras. Los métodos de incrustación de palabras abordan este problema, pero están limitados por la necesidad de un enorme corpus de conjuntos de datos de texto para el entrenamiento. Como resultado, los científicos prefieren utilizar vectores de incrustación de palabras previamente entrenado. Sin embargo, este enfoque no puede funcionar para las palabras que faltan en estos corpus de datos de texto.

2.6. Clasificación de Textos Multi-etiquetados

El interés por la clasificación automática de textos multi-etiqueta ha ido aumentando en los últimos años. Entre las propuestas presentadas por [32], el enfoque más usado es el denominado transformación del problema. En un problema multi-etiqueta existe un número de etiquetas finito $L = \{\lambda_j : j = 1 \dots l\}$, donde λ_j corresponde a la etiqueta j -ésima, y al set de documentos etiquetados $D = \{f\{\mathbf{x}_i, \mathbf{Y}_i : i = 1 \dots d\}\}$, donde \mathbf{x}_i representa el vector de características y $\mathbf{Y}_i \subset L$ es el conjunto de etiquetas del texto i -ésimo.

El cuadro 2.1 presenta un ejemplo de textos multi-etiquetados. En ella se representa que el Texto 1 pertenece a las clases ‘Deportes’ y ‘Política’, que el Texto 2 pertenece a las clases ‘Ciencias’ ‘Política’, que el Texto 3 pertenece solo a la clase ‘Deportes’ y finalmente que el Texto 4 pertenece a las clases ‘Religión’ y ‘Ciencias’.

Etiquetas (l) Textos (d)	Deportes	Religión	Ciencias	Política
Texto 1	x			x
Texto 2			x	x
Texto 3	x			
Texto 4		x	x	

CUADRO 2.1. Representación de un Conjunto de Ejemplos Multi-Etiquetados

Los métodos para resolver este problema se encuentran agrupados en dos enfoques, los cuales son: transformación del problema y adaptación del modelo [32]. El enfoque de transformación del problema es independiente del algoritmo, es decir, transforma la tarea de aprendizaje de multi-etiqueta en una tarea de clasificación de una sola etiqueta. De esta forma, este método puede ser implementado usando algoritmos existentes. El método más común de transformación del problema, llamado Relevancia Binaria (*Binary Relevance, BR*) aprende $|L|$ clasificadores binarios $H_{(\lambda_j)} : X \rightarrow \{\lambda_j, \neg\lambda_j\}$, uno para cada etiqueta diferente λ_j en L . A través del uso de Relevancia Binaria se transforma el conjunto de datos original en $|L|$ conjuntos de datos D_{λ_j} . Cada D_{λ_j}

etiqueta cada ejemplo de texto en D con λ_j si es que λ_j es contenida en el ejemplo o $\hat{A} \neg \lambda_j$ si el texto de ejemplo no contiene la etiqueta. BR brinda la misma solución, tanto para problemas de una sola etiqueta, como para problemas multi-etiqueta usando clasificadores binarios. Para la clasificación de una nueva instancia x , este método genera un set de etiquetas como la unión de las etiquetas generadas por el clasificador $|L|$, $H_{BR}(x) = \cup_{(\lambda_j \in L)} \{\lambda_j\} : H_{\lambda_j}(x) = \lambda_j$. Esta suele ser la transformación más común y es la misma solución utilizada cuando se pretende lidiar con un problema de clasificación multi-clase usando clasificadores binarios (ver cuadro 2.2).

Etiqueta \ Textos	Deportes	No deportes	Etiqueta \ Textos	Religión	No religión
Texto 1	X		Texto 1		X
Texto 2		X	Texto 2		X
Texto 3	X		Texto 3		X
Texto 4		X	Texto 4	X	

Etiqueta \ Textos	Política	No política	Etiqueta \ Textos	Ciencias	No ciencias
Texto 1	X		Texto 1		X
Texto 2	X		Texto 2	X	
Texto 3		X	Texto 3		X
Texto 4		X	Texto 4	X	

CUADRO 2.2. Transformación Resultante utilizando Relevancia Binaria

Otro tipo de transformación del problema se denomina Conjunto Potencia de Etiquetas (*Label Powerset*, LP). En esta transformación cada conjunto de etiquetas se considera una nueva categoría. Luego, si tenemos LP combinaciones, es posible utilizar $|LP|$ clasificadores binarios, uno para cada nueva etiqueta. El conjunto de datos se maneja como uno del tipo etiqueta simple para luego construir un clasificador de etiqueta simple con múltiples clases disjuntas (ver cuadro 2.3).

Ejemplo \ Etiqueta	Deportes	Deportes y Política	Ciencias y Política	Ciencias y Religión
Texto 1		x		
Texto 2			x	
Texto 3	x			
Texto 4				x

CUADRO 2.3. Transformación del Problema Usando *Label Powerset*

El segundo método se ocupa de adaptar algunos modelos y algoritmos de aprendizaje específicos para que puedan manejar datos de multi-etiquetados directamente. Estas adaptaciones son logradas gracias a ajustes de los modelos, tales como modificaciones a formulaciones clásicas de estadísticas o teoría de la información. El pre-procesamiento de los documentos para lograr una mejor representación también puede ser considerado dentro de este tipo de transformación.

La clasificación automática de textos multi-etiqueta ha sido abordada también por medio de algoritmos, los cuales capturan directamente las características del problema de multi-etiqueta. [33] proponen un método con base en lógica difusa, donde un texto con multi-etiquetado puede pertenecer a una, o más de una categoría. Los autores exponen que al incorporar técnicas difusas, el método puede sobreponerse a los problemas causados por los altos requerimientos de memoria o bajo rendimiento. [34] se enfoca en resolver las limitaciones que tiene el algoritmo de *Backpropagation Learning* para que pueda trabajar con datos multi-etiquetados y propone el *Backpropagation Multi-Label Learning (BP-MLL)*, en esta propuesta se utiliza un enfoque de red neuronal muy simple para tareas de clasificación de textos multi-etiquetados a gran escala.

[35] propone un modelo basado en redes de atención de grafos para capturar la estructura de dependencia entre las etiquetas (MAGNET). Los resultados del modelo propuesto se validan en cinco conjuntos de datos MLTC del mundo real. El modelo propuesto logra un rendimiento similar o mejor en comparación con los modelos anteriores de última generación. La estructura del modelo MAGNET utiliza grafos de capas de atención para la clasificación para generar las entradas para que una BiLSTM genere los vectores de características que se codifican

mediante incrustaciones del modelo BERT [36]. La entrada para el grafo de la red de atención es una matriz de adyacencia y los vectores de etiqueta. La salida GAT son características de etiquetas que se aplican a los vectores de características obtenidos con BiLSTM. Utiliza métricas de desempeño que no permite evaluar su desempeño en conjuntos de datos desbalanceados. La propuesta de MAGNET mejoró en desempeño marginalmente respecto a los modelos tradicionales, tales como Binary Relevance y Chain Classifiers.

El método Label-Wise Pre-Training (LW-PT) para obtener una representación de documentos con información de las etiquetas es propuesto en [37]. Para esto, un documento de múltiples etiquetas se representa como una combinación de múltiples representaciones de etiquetas, y se asume que las etiquetas correlacionadas siempre co-ocurren en documentos similares. LW-PT implementa esta idea mediante la construcción de clasificadores de documentos por etiquetas y entrena codificadores de documentos por etiquetas. El codificador de etiquetas preentrenado se ajusta con precisión. Los resultados experimentales validan el método propuesto en dos set de datos y se comparan con técnicas tradicionales solamente en algunas métricas de evaluación.

Por otra parte, [38] propone un Modelo Global Jerárquico para Clasificación de Textos Multi-Etiquetados que busca aprovechar las relaciones de dependencia entre las etiquetas. En línea con la explotación del Aprendizaje Profundo, [39] plantea un modelo para la clasificación de texto de etiquetas múltiples extremas (eXtreme Multilabel Text Classification), enfrentando el problema de asignar a cada documento el subconjunto más relevante de etiquetas de clase de una colección de etiquetas extremadamente grande, donde el número de etiquetas podría alcanzar a cientos de miles o millones. También [40] se utiliza una estimación de la distribución de las categorías en un espacio no lineal de embedding en un modelo que llaman Prototypical Networks for Multi-label Learning (PNML) y en [35] se propone Classification Neural Networks (CNN), diferente de Redes Neuronales Convolucionales, como un nuevo enfoque basado en aprendizaje profundo para enfrentar el problema de jerarquía de etiquetas.

Sin importar el enfoque de soluciones al problema de multi-etiqueta y los algoritmos que lo resuelvan, según [10], cualquier tarea de clasificación de texto tiene complejidades debido a que el espacio de características es altamente multidimensional, un uso heterogéneo de los términos y un alto nivel de redundancia. Los problemas multi-etiqueta tienen complejidades adicionales, incluyendo un gran número de etiquetas y el desbalance de las mismas a través del conjunto de documentos.

Aunque las medidas tradicionales de evaluación del desempeño de los clasificadores automáticos, tal como las medidas *macro-F* y *Hamming Loss* son útiles en casos multi-etiqueta, también han surgido nuevas medidas de evaluación con la intención de analizar el desempeño en la asignación del conjunto de etiquetas que le corresponden a cada documento, tal como la exactitud del conjunto de etiquetas, medida denominada *Label-Set Accuracy* [41].

En general, las propuestas basadas en modelos de aprendizaje profundo se utilizan para conjuntos de datos mas grandes, y explotan las altas capacidades tecnológicas disponibles para identificar patrones en las relaciones entre etiquetas. Es escasa la literatura donde se realicen evaluaciones comparativas de modelos superficiales y profundos en los mismos conjuntos de textos multi etiquetados.

En resumen, en favor de los métodos basados en aprendizaje superficial se puede decir que la clasificación basada en aprendizaje profundo tiene una alta complejidad computacional y varias aproximaciones requieren de conocimientos o experiencia previa en el dominio para extraer características del texto original. Desde la perspectiva de las aplicaciones de máquinas de aprendizaje, y de su creciente tasa de adopción en la industria, hay que considerar la necesidad de desarrollar modelos livianos computacionalmente que puedan implementarse bajo condiciones tecnológicas asequibles para compañías de diferentes tamaños.

También se puede observar que, para conjuntos de datos pequeños, los modelos de aprendizaje superficial generalmente presentan un mejor rendimiento que los modelos de aprendizaje profundo bajo la limitación

de la complejidad computacional. Esto puede ser más relevante cuando el número promedio de etiquetas asociadas con cada documento u objeto es mayor.

Finalmente, interpretabilidad de los modelos de aprendizaje profundo, no es comparable con la de los modelos superficiales. Esto último limita las aplicaciones donde es necesario interpretar el funcionamiento de los modelos para explicar por qué y cómo funcionan bien. Este es un factor limitante para los casos de uso que requieren explicaciones de las características involucradas en el modelado y muchos modelos prescriptivos aplicados al área de salud o en automatizaciones que pueden afectar la libertad de las personas. Lo anterior también es necesario para obtener retroalimentación acerca de las características que ayudan a discriminar para hacer recomendaciones de modificación a los objetos para evaluación.

Capítulo 3

Marco Metodológico

3.1. Conjuntos de Datos

Existen muchos conjuntos de datos estandarizados para hacer probar los modelos, los principales diez conjuntos de datos textuales multi-etiquetados son: REUTERS- 21578, OHSUMED, ENRON, SLASH-DOT, LANGLOG, BIBTEX, TMC, Yahoo Education, Yahoo Science y MEDICAL. Para REUTERS-21578, que es un conjunto de textos de noticias, se consideró un subconjunto modificado que se propuso en [41] con el fin de poder obtener medidas de desempeño comparativas. El conjunto de datos OHSUMED es una partición de la base de datos MEDLINE, que es una biblioteca de artículos científicos publicados en revistas médicas. La colección OHSUMED también se ha reducido de 50.216 a 13.929 textos. Este subconjunto contiene las 10 categorías más representativas de las 23 categorías originales. El conjunto de datos de Enron es una colección de textos creados por el proyecto CALO (*Cognitive Assistant that Learns and Organizes*), que contiene 1.702 mensajes de correo electrónico y 52 categorías. Por último, el conjunto de datos Medical fue creado por la *Computational Medicine Center, 2007* a propósito del *Language Processing Challenge, 2007*, contiene 978 textos clínicos de informes de radiología y considera 45 categorías de códigos médicos. TMC2007 es un subconjunto del conjunto de datos del Sistema de informes de seguridad operacional de la aviación. Finalmente, usamos páginas web reales vinculadas desde el dominio "yahoo.com", específicamente comparamos "Science" y "Education". El cuadro 2.4 presenta las características del conjunto de datos pre-procesados.

3.2. Medidas de desempeño

Las medidas tradicionales de evaluación como la medida F y la *Hamming Loss* son útiles en el caso de conjuntos multi-etiquetados. Sin embargo, como ya hemos planteado, existen nuevas medidas destinadas a evaluar el desempeño en la asignación del conjunto de etiquetas,

Data Set	Núm de Etiquetas	Núm de Docs	Tamaño del Voc	Card	Dens	Div
Enron	52	1.702	1.001	3,378	0,064	0,442
Bibtex	159	7.395	1.836	2,402	0,015	0,386
TMC2007	22	28.600	49.060	2,158	0,098	0,047
Ohsumed	23	13.929	1.002	1,663	0,072	0,082
Education	33	12.030	27.530	1,463	0,044	0,042
Reuters	103	6.000	500	1,462	0,014	0,135
Science	40	6.428	37.490	1,450	0,036	0,071
Medical	45	978	1.449	1,245	0,028	0,096
Slashdot	22	3.782	1.079	1,181	0,054	0,041
Langlog	75	1.460	1.004	1,180	0,016	0,208

CUADRO 3.1. Características del Conjunto de Datos Pre-procesados. Cardinalidad (Card) mide el número promedio de etiquetas asociadas con cada documento. Densidad (Dens) se define como la cardinalidad dividida por la cantidad de etiquetas. La Diversidad (Div) representa el porcentaje de conjuntos de etiquetas presentes en el conjunto dividido por el número de conjuntos de etiquetas posibles. El Tamaño del Vocabulario considera el volumen de palabras distintas.

como lo es la precisión del conjunto de etiquetas (*Label-Set Accuracy*).

Para describir las medidas de desempeño, se utiliza la siguiente notación: considerando el vector $\mathbf{Y}_i \in [0, 1]^{|L|} : i = 1 \dots d$, entonces cada etiqueta será relevante si $y_{i,j} = 1$, y por su parte, la predicción del clasificador automático será $y'_{i,j} = 1$, donde d es el número de documentos y $|L|$ es el número de posibles etiquetas.

Basándose en la notación anterior, *Hamming Loss* se define como en la Ecuación 2.7:

$$(3.1) \quad \text{Hamming - Loss} (\mathbf{Y}, \mathbf{Y}') = \frac{1}{d} \frac{1}{|L|} \sum_{i=1}^d \sum_{j=1}^{|L|} |y'_{i,j} \Delta y_{i,j}|$$

donde, $y'_{i,j} \Delta y_{i,j}$ representa la diferencia de etiquetas asignadas por el clasificador y las etiquetas reales. Esta medida busca medir la diferencia entre cada etiqueta que los textos realmente tienen, con cada etiqueta que asignó el clasificador automático a dichos textos. Mientras más bajo es el valor obtenido, mejor es el desempeño.

Otra medida multi-etiqueta es la precisión del conjunto de etiquetas (*Accuracy*) y es definida como en la Ecuación 2.8:

$$(3.2) \quad \text{Accuracy}(\mathbf{y}, \mathbf{y}') = \frac{1}{d} \sum_{i=1}^d 1(y_i = y'_i)$$

Esta media de desempeño permite medir, para cada texto, las etiquetas correctamente asignadas. En la clasificación de etiquetas múltiples, la función devuelve la precisión del subconjunto. Si todo el conjunto de etiquetas predichas para una muestra coincide estrictamente con el conjunto real de etiquetas, entonces la precisión del subconjunto es 1; de lo contrario, es 0. Mientras más alto es el valor obtenido, mejor es el desempeño.

La medida F , comúnmente utilizada en recuperación de información, es muy popular en clasificación de textos multi-etiquetados. La medida F es la media armónica entre precisión y exhaustividad (*recall*). La medida F (F_1) para cada etiqueta se calcula como se muestra en la Ecuación 2.9:

$$(3.3) \quad F_1(Y_i, Y'_i) = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

donde la precisión es la fracción de las predicciones que son realmente relevantes y *recall* es la fracción de relevancia real con respecto a las *predicciones*. Mientras más alto es el valor de F , mejor es el desempeño.

Para el caso multi-etiqueta es necesario combinar los diferentes F_1 de cada evaluación de la etiqueta. Para eso usamos macro- F_1 que es el promedio del F_1 para cada etiqueta.

Capítulo 4

Propuesta

Si bien en los últimos años se ha producido un crecimiento del interés de la comunidad científica en los modelos de aprendizaje profundo, existe un debate entre las aproximaciones de “caja blanca” vs “caja negra”, con ventajas y desventajas para ambos enfoques. Del aprendizaje superficial se destaca la factibilidad de trabajar con conjuntos relativamente pequeños de datos y la interpretabilidad. De los modelos de aprendizaje profundo se destaca su buen desempeño y su robustez. Por lo tanto, no es posible descartar totalmente las técnicas y modelos basados en aprendizaje superficial, sobretodo cuando el conjunto de casos de entrenamiento no tiene un gran volumen de datos y el conjunto de características no es muy extenso.

Como ya se ha mencionado, en esta aproximación se planea utilizar métodos de clasificación basados en aprendizaje superficial combinando modificación de la representación con transformación del problema. Aunque en esta investigación no se utilizan métodos de clasificación basados en aprendizaje profundo, tales como BERT y sus variantes, también sería posible utilizar esta aproximación para modificar las características de la representación de los textos originales y generar nuevas capas o entradas a métodos de aprendizaje profundo. Sobretodo, cuando se trabaje con conjuntos de datos pequeños. Esto también otorgaría mejor interpretabilidad a los modelos de aprendizaje profundo.

Por otra parte, en el caso del problema de textos multi-etiquetados, los modelos de clasificación deben lidiar con conjuntos de datos con una alta cardinalidad, densidad y diversidad de etiquetas. La cardinalidad mide el número promedio de etiquetas asociadas con cada documento, la densidad es la cardinalidad dividida por la cantidad de etiquetas y la diversidad representa el porcentaje de conjuntos de etiquetas presentes en el corpus dividido por el número de conjuntos de etiquetas posibles.

En este trabajo, utilizamos conjuntos de datos con cardinalidad entre 1,18 y 3,28; densidad entre 0,014 y 0,098; y diversidad entre 0,041 y 0,442.

En esta sección se explica la bien conocida representación *tf-idf* [30] y se presenta nuestra propuesta de función para la ponderación de términos *rfl*. Sobre la base de esta última, proponemos dos nuevas representaciones, una basada en el modelo Bernoulli Multivariado llamada *bin-rfl* y otra basada en el modelo Multinomial *tf-rfl*. En este tipo de representación el indicador $f_{t,d}$ puede tener valores entre cero y uno ($[0,1]$) es llamado Modelo Multinomial por [27], y es diferente al Modelo Bernoulli Multi-variado, donde el indicador es $bin_{t,d}$, el cual es representado por uno cuando el término t existe al menos una vez en el documento d , es decir, puede tener valor cero o uno ($\{0,1\}$). El factor basado en el Modelo Bernoulli Multi-variado es llamado representación Binaria o modelo Booleano. Muchos problemas, ya sea por su naturaleza o por las medidas que se pueden obtener de ellos, utilizan el modelo de representación basado en el Modelo Bernoulli Multivariado.

Se plantea con ello la hipótesis de que una modificación supervisada a la representación del texto que considere representaciones de frecuencia o representaciones binarias, junto con una función para la ponderación supervisada de los términos que está basada en los ejemplos conocidos, según sus etiquetas, podría mejorar significativamente el rendimiento de los clasificadores. Para el método de ponderación de términos para problemas de múltiples etiquetas usaremos como variables las descritas en el cuadro 3.1: a_{t,λ_j} , la cual representa el número de documentos en la categoría λ_j que contiene el término t y d_{t,λ_j} que representa el número de documentos en la categoría λ_j que no contiene el término t .

Etiqueta (l)	Deportes (1)	Religión (2)	Ciencias (3)	Política (4)
t (contiene el término)	$a_{t,1}$	$a_{t,2}$	$a_{t,3}$	$a_{t,4}$
\bar{t} (no contiene el término)	$d_{t,1}$	$d_{t,2}$	$d_{t,3}$	$d_{t,4}$

CUADRO 4.1. Variables utilizadas para ponderar en un problema multietiqueta dado un término t y 4 categorías

4.1. Representación Term Frequency-Inverse Document Frequency (*tf-idf*)

Según [24], la representación de textos más utilizada para clasificación de textos es *tf-idf* de [30]. Donde, cada componente del vector es calculado según la Ecuación 3.1:

$$(4.1) \quad tf - idf_{td} = f_{t,d} \times \log_{10} \left(\frac{N}{N_t} \right),$$

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

La principal contribución de esta representación es que pondera con menor importancia los términos que son muy frecuentes en la colección de documentos a través del factor N/N_t .

4.2. Representación Term Frequency-Relevance Frequency for a Label (*tf-rfl*)

En la investigación realizada por [1] se presentaron resultados preliminares de la representación *Relevance frequency for a label*, *tf-rfl*. Esta representación se describe en la siguiente ecuación, como una nueva representación para problemas multi-etiqueta.

$$(4.2) \quad tf - rfl_{tdl} = f_{t,d} \times \log_2 \left(2 + \frac{a_{t,l}}{\max(1, \text{mean}(a_{t,\lambda_{j/l}}))} \right),$$

donde $f_{t,d}$ es la frecuencia del término t en el documento d , $a_{t,l}$ es el número de documentos bajo la categoría en evaluación l que contienen el término t , y $\text{mean}(a_{t,\lambda_{j/l}})$ es el promedio del número de documentos que contienen el término t entre el conjunto de documentos etiquetados en otra categoría diferente de l , es decir $a_{t,\lambda_{j/l}} = \{a_{t,\lambda_1}, \dots, a_{t,\lambda_{l-1}}, a_{t,\lambda_{l+1}}, \dots, a_{t,|L|}\}$.

El valor constante 2 en el lado derecho de la fórmula se asigna porque la base de la operación logarítmica es 2. Sin la constante 2, podría

tener el efecto de dar valor cero a otros términos. Se podrían utilizar otras bases para la función de logaritmo, lo que implicaría también modificar el valor de este parámetro.

La principal contribución de esta representación es que pondera con menor importancia los términos que son igualmente frecuentes en las diferentes categorías y pondera con mayor importancia los términos que son más frecuentes en la categoría bajo evaluación.

También es posible utilizar *bin - idf* basado en el modelo de múltiples variables Bernoulli en lugar de *tf - idf* basado en el modelo Multivariado. En este caso, en lugar de utilizar $f_{t,d}$ se utiliza $bin_{t,d}$.

Con el fin de evaluar la mejora del desempeño debido al uso de la ponderación *rfl*, este trabajo presentará una nueva representación basada en la aparición de términos en cada documento, es decir, representación binaria o representación booleana. Esta representación, basada en el Modelo de Bernoulli Multivariado, utiliza menos información que la basada en el Modelo Multinomial, ya que solo se utiliza la información de existencia o no de una palabra en el texto y no su frecuencia de aparición.

4.3. Representación Multi-Variate Bernoulli Model - Label Dependent (*bin-rfl*)

La nueva representación para el problema de etiquetado múltiple, que se propone en este trabajo, llamado *bin - rfl*, se basa en una representación del modelo de Bernoulli multivariado que se pondera usando el término frecuencia de una etiqueta y se calcula como en la Ecuación número 4.3:

$$(4.3) \quad bin - rfl_{tdl} = bin_{t,d} \times \log_2 \left(2 + \frac{a_{t,1}}{\max(1, \text{mean}(a_{t,\lambda_j/l}))} \right),$$

donde $bin_{t,d}$ toma el valor 1, si el término t está presente en el documento d y 0, si el término t no está presente en el documento d , $a_{t,l}$ es el número de documentos de la categoría bajo evaluación que contiene

el término t , y $mean(a_{t,\lambda_{j/l}})$ es el número promedio de documentos que contienen el término t para cada el conjunto de documentos etiquetados distinto de l . Esta nueva representación ayuda a hacer una mejor distinción de los términos, que se refleja en una mejor clasificación de rendimiento, como se verá en la sección resultados.

4.4. Interpretación Probabilística

Una interpretación probabilística de esta representación es que $f_{t,d}$ es una estimación de $P(t_i/d_j)$, es decir, de la probabilidad que el término i esté en la colección de documento j . Asimismo, la ponderación $idf = \log_{10}(N/N_t)$ es función de la $1/P(t_i/N)$, es decir, de la probabilidad de que el término i esté en el conjunto de documentos, es decir, $P(t_i)$. Entonces, la función $tf - idf$ está dada por:

$$(4.4) \quad tf - idf_{td} = \frac{P(t_i/d_j)}{\log(P(t_i/N))}.$$

Note que el factor de ponderación idf no toma en cuenta que los documentos pueden tener varias categorías.

Para el caso la función de ponderación rfl , se puede considerar que es una estimación de la $P(t_i/N_l)/P(t_i/N_{j/l})$, es decir, de la probabilidad de que el término i esté en el conjunto de documentos etiquetados bajo la etiqueta l , sobre la probabilidad de que esté en el conjunto de documentos etiquetados en otras etiquetas, diferentes de l . Entonces, la función $tf - rfl$ se puede representar como:

$$(4.5) \quad tf - rfl_{tdl} = \frac{P(t_i/d_j)}{\log\left(\frac{P(t_i/N_l)}{P(t_i/N_{j/l})}\right)}.$$

Con esto, se plantea una función rfl de ponderación de términos que enfrenta el problema de clasificación con múltiples etiquetas, cosa que no considera idf .

4.5. Interpretación como *Ensemble*

Los métodos de *Ensemble* (o Comités de Máquinas) se han utilizado ampliamente en el campo del aprendizaje de máquinas, debido a que mejoran el desempeño de las tareas de clasificación mediante la combinación de máquinas similares o diferentes, sus datos de entrenamiento

y la incorporación de diversidad en el método de aprendizaje. Se ha demostrado teórica y empíricamente que los métodos de *Ensemble* son más precisos que los clasificadores simples [42].

Siguiendo la taxonomía propuesta por [43], se puede plantear que nuestra propuesta introduce mayor diversidad a través de dos rutas. Primero, a partir de la manipulación del conjunto de entrenamiento se incorpora información del dominio de las etiquetas para cada miembro del *Ensemble* procese diferentes entradas manipuladas previamente. Y segundo, a partir de especializar a cada uno de los miembros del comité de clasificadores en cada una de las etiquetas l del conjunto de entrenamiento.

La representación rfl modifica los conjuntos de entrenamiento incorporando información sobre las características que diferencian los conjuntos de ejemplos de diferentes etiquetas. A su vez, cada clasificador utiliza estos ejemplos mediante clasificaciones binarias de cada etiqueta l : pertenece o no pertenece.

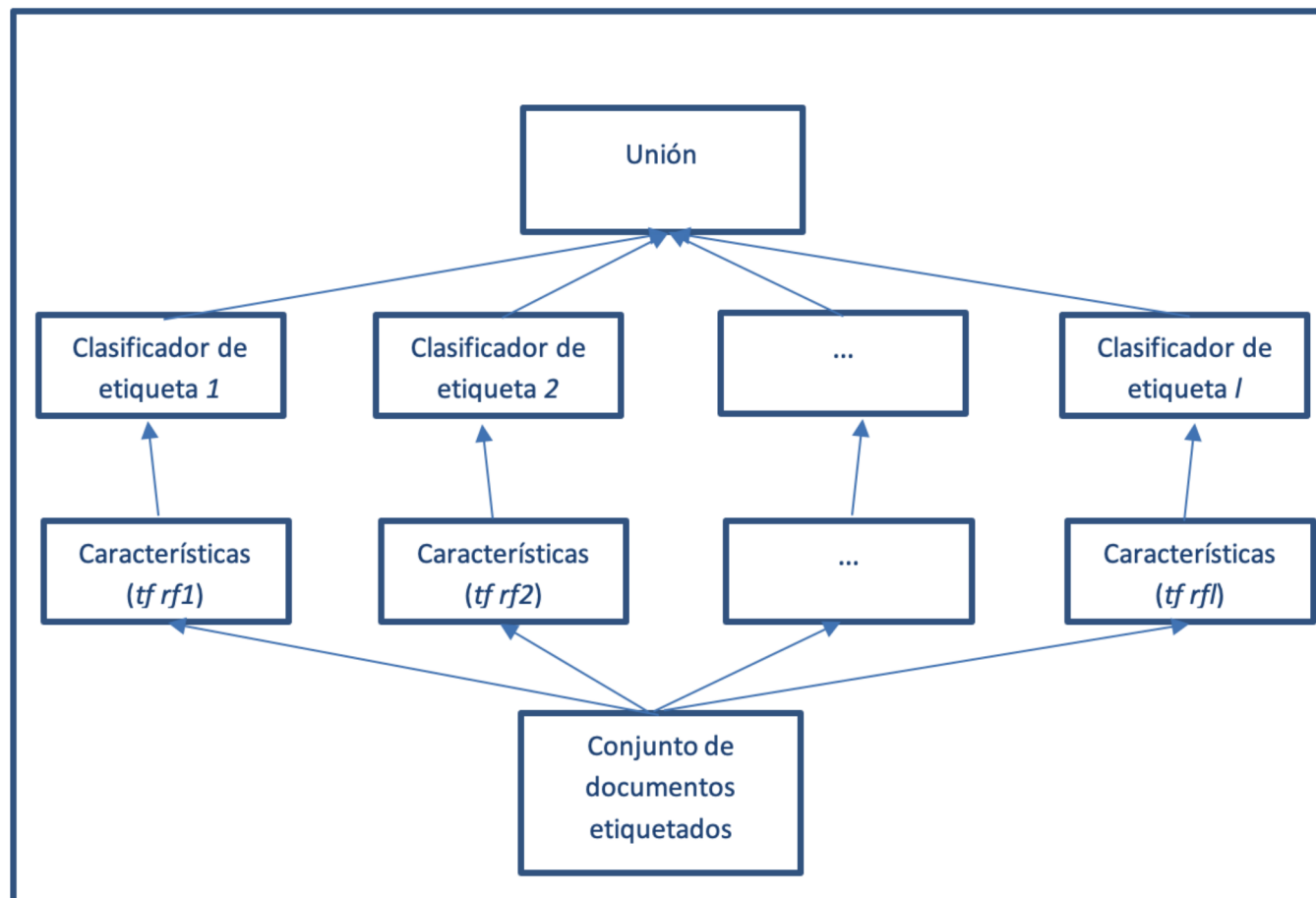
El siguiente esquema representa nuestra propuesta como un *Ensemble*:

4.6. Interpretación Geométrica

Si nos basamos en el *Vector Space Model*, podemos interpretar que cada característica de un documento es representada como una dimensión del vector de características.

La modificación rfl considera la relación entre las características de los documentos que pertenecen a la etiqueta bajo clasificación y las características de los documentos que pertenecen a otras etiquetas, diferentes de la etiqueta l bajo evaluación.

Al aplicar el ponderador $a_{t,l}/\mu(a_{t,\lambda_{j/l}})$, se busca que el valor correspondiente en el vector a la características t se incremente cuando esa característica ocurra más en el conjunto de documentos etiquetados bajo l que en el resto de las etiquetas, y a su vez, que disminuya cuando t ocurra menos en l que en el resto de las etiquetas.

FIGURA 4.1. Propuesta como un *Ensemble*

La interpretación geométrica es que el valor en la dimensión t del vector aumenta, es decir, se aleja de los otros vectores correspondientes a las otras etiquetas. Esto se ejemplifica en la siguiente gráfica:

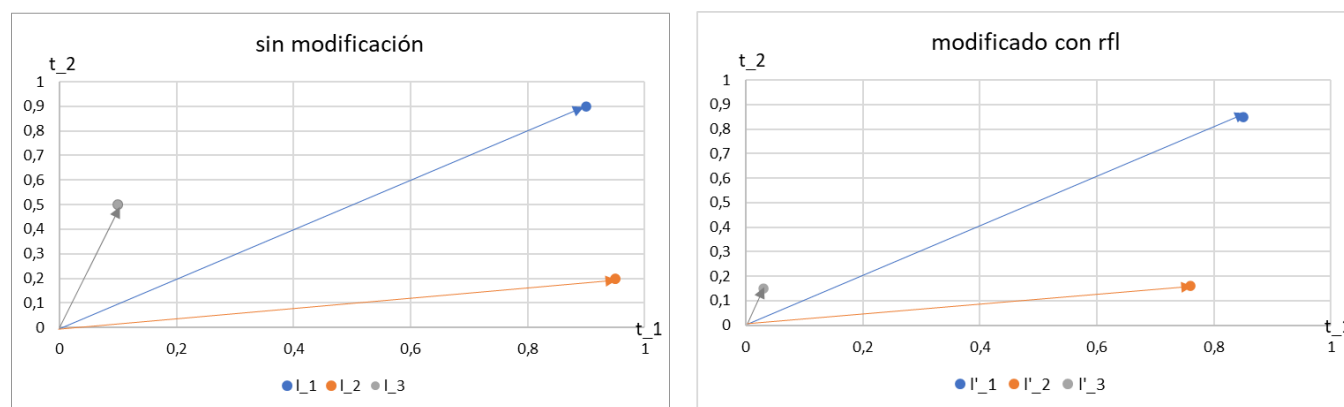


FIGURA 4.2. Interpretación Geométrica

		t_1	t_2	$a_{t,1}$	$media(a_t, \lambda_{j/l})$	rfl	t'_1	t'_2
l_1	$d_1 = 5$	0,90	0,90	5	60,0	0,94	0,85	0,85
l_2	$d_1 = 20$	0,95	0,2	20	52,2	0,80	0,76	0,16
l_3	$d_1 = 100$	0,10	0,50	100	12,5	0,30	0,03	0,15

4.7. Método de Clasificación

Los métodos basados en aprendizaje superficial, tales como Redes Neuronales Artificiales Feedforward (FANN), Máquinas de Soporte Vectorial (SVM) y Naive Bayes (NB), tienen una menor complejidad que la clasificación basada en aprendizaje profundo y varias aproximaciones de estos últimos modelos, tales como BERT y sus variantes, requieren de conocimientos o experiencia previa en el dominio para extraer características del texto original. A su vez, que los conjuntos de datos utilizados tradicionalmente en la literatura para evaluar el desempeño de modelos en el problema de clasificación multi etiqueta ya han sido preprocesados de modo estándar y normalmente se utilizan esas matrices de textos postprocesados para realizar las comparaciones bajo las mismas condiciones de preprocesamiento. También se puede observar que, para conjuntos de datos pequeños, los modelos de aprendizaje superficial presentan desempeños variables con respecto a los modelos de aprendizaje profundo. Asimismo, que este aspecto puede ser más relevante cuando el número promedio de etiquetas asociadas con cada documento u objeto es mayor.

La propuesta del método de ponderación de términos incluye información de la frecuencia de ocurrencia de un término t en cada grupo de los documentos etiquetados con otras etiquetas distintas de las que están bajo evaluación. Con rfl , se espera que $mean(a_{t,\lambda_{j/l}})$ sea mayor si el término t aparece más frecuentemente en documentos con etiqueta $\lambda_j = l$ que en documentos con otras etiquetas $\lambda_{j/l}$. Asimismo, tendrá un valor más bajo si el término t es más frecuente en documentos con etiquetas que no sean l . De este modo, la ponderación rfl resulta ser un mejor discriminador entre categorías.

En este trabajo se propone una representación basada en $bin-rfl$ y $tf-rfl$, junto a clasificadores binarios, utilizando las transformaciones del problema: *Relevancia Binaria y Label Powerset*. Para ello se transforma el problema de etiquetado múltiple en problemas binarios, y luego para cada documento d se construyen las representaciones $bin-rfl$ para cada etiqueta y se clasifican usando clasificadores binarios. Se debe destacar que cada documento está representado con un vector diferente cuando está bajo evaluación de cada etiqueta, porque el factor de

ponderación depende de la etiqueta bajo evaluación.

Resultados Experimentales

5.1. Resultados

Con el objetivo de comparar el efecto de utilizar la función *rfl* para modificar la representación, realizamos diferentes experimentos de clasificación utilizando diez conjuntos diferentes de set de datos ampliamente trabajados en la literatura (Reuters, Ohsumed, Enron, Slashdot, Langdot, Bibtex, Medical, TMC2007, Science y Education), utilizando representaciones *bin-rfl* y *bin-idf*, ambas con transformaciones de Relevancia Binaria (*Binary Relevance*) y la *Label Powerset*, y dos clasificadores lineales diferentes (SVM y ANN). El objetivo de utilizar estos clasificadores lineales es evaluar la modificación, independiente del modelo de clasificación.

Se puede evaluar el impacto de la modificación de la representación utilizando modelos de aprendizaje superficial, que funcionan bien bajo condiciones de complejidad computacional limitadas, ya que no requieren de conocimientos o experiencia previa en el dominio para extraer características del texto original. A su vez, se utilizan conjuntos de datos del problema de clasificación multi etiqueta, de menor tamaño, que son ampliamente conocidos en la literatura. Estos conjuntos ya han sido preprocesados de modo estándar.

Los cuadros 5.1 a 5.6 muestran los diferentes métodos y su desempeño en términos de diferentes medidas de desempeño descritas previamente.

TP	Repr.	Reuters	Ohsumed	Enron	Slashdot	Langlog
BR	<i>bin-idf</i>	0,247	0,402	0,080	0,257	0,240
BR	<i>bin-rfl</i>	0,200	0,414	0,105	0,325	0,334
LP	<i>bin-idf</i>	0,353	0,381	0,163	0,424	0,313
LP	<i>bin-rfl</i>	0,350	0,401	0,212	0,442	0,519

TP	Repr.	Bibtex	Medical	TMC2007	Science	Education
BR	<i>bin-idf</i>	0,122	0,577	0,220	0,290	0,242
BR	<i>bin-rfl</i>	0,138	0,413	0,216	0,293	0,171
LP	<i>bin-idf</i>	0,230	0,628	0,126	0,342	0,281
LP	<i>bin-rfl</i>	0,300	0,681	0,243	0,526	0,451

CUADRO 5.1. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con **SVM** en términos de *Accuracy*

TP	Repr.	Reuters	Ohsumed	Enron	Slashdot	Langlog
BR	<i>bin-idf</i>	0,298	0,388	0,097	0,302	0,240
BR	<i>bin-rfl</i>	0,373	0,395	0,225	0,416	0,334
LP	<i>bin-idf</i>	0,358	0,373	0,122	0,352	0,313
LP	<i>bin-rfl</i>	0,462	0,388	0,212	0,489	0,519

TP	Repr.	Bibtex	Medical	TMC2007	Science	Education
BR	<i>bin-idf</i>	0,172	0,573	0,265	0,286	0,297
BR	<i>bin-rfl</i>	0,214	0,584	0,282	0,385	0,261
LP	<i>bin-idf</i>	0,138	0,537	0,175	0,370	0,310
LP	<i>bin-rfl</i>	0,245	0,663	0,323	0,508	0,488

CUADRO 5.2. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con **ANN** en términos de *Accuracy*

Acerca de los clasificadores, los resultados de las SVM son reñidos con los de las ANN. Relevancia Binaria en general tiene un mejor rendimiento que la *Label Powerset*, a menos que la evaluación sea en términos de *Accuracy*, donde algunos conjuntos de datos LP tienen un mejor rendimiento que BR.

TP	Repr.	Reuters	Ohsumed	Enron	Slashdot	Langlog
BR	<i>bin-idf</i>	0,017	0,064	0,065	0,063	0,018
BR	<i>bin-rfl</i>	0,018	0,063	0,050	0,054	0,017
LP	<i>bin-idf</i>	0,017	0,060	0,056	0,052	0,020
LP	<i>bin-rfl</i>	0,019	0,059	0,057	0,055	0,017
TP	Repr.	Bibtex	Medical	TMC2007	Science	Education
BR	<i>bin-idf</i>	0,018	0,014	0,073	0,037	0,048
BR	<i>bin-rfl</i>	0,018	0,021	0,076	0,049	0,063
LP	<i>bin-idf</i>	0,016	0,014	0,156	0,057	0,068
LP	<i>bin-rfl</i>	0,019	0,014	0,124	0,040	0,051

CUADRO 5.3. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con **SVM** en términos de *Hamming Loss*

TP	Repr.	Reuters	Ohsumed	Enron	Slashdot	Langlog
BR	<i>bin-idf</i>	0,015	0,067	0,057	0,055	0,016
BR	<i>bin-rfl</i>	0,013	0,064	0,039	0,042	0,012
LP	<i>bin-idf</i>	0,016	0,062	0,056	0,062	0,027
LP	<i>bin-rfl</i>	0,014	0,060	0,062	0,051	0,022
TP	Repr.	Bibtex	Medical	TMC2007	Science	Education
BR	<i>bin-idf</i>	0,013	0,013	0,064	0,027	0,040
BR	<i>bin-rfl</i>	0,014	0,013	0,063	0,029	0,047
LP	<i>bin-idf</i>	0,023	0,020	0,100	0,046	0,053
LP	<i>bin-rfl</i>	0,020	0,014	0,077	0,038	0,040

CUADRO 5.4. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con **ANN** en términos de *Hamming Loss*

Acerca de la representación, en casi todos los casos, la representación *bin-rfl* presenta mejoras a *bin-idf*. Como se muestra en los cuadros 5.1 y 5.2, se obtiene una mejora promedio superior al 15% (con SVM) y al 40% (con ANN) en términos de *Accuracy*. Del mismo modo, se obtienen mejoras del 12% en términos de *Hamming-Loss* con ANN, como se muestra en los cuadros 5.3 y 5.4.

En [35] se se utiliza el modelo Classification Neural Networks (CNN) y se prueba en los sets de datos: Enron, Medical y Science, entre otros.

A partir de esto, podemos comparar los resultados en términos de *Hamming-Loss*. Se reporta un valor de 0,046 para Enron; 0,013 para Medical y 0,031 para Science. En esta comparación, nuestra propuesta obtiene un valor de *Hamming-Loss* de 0,039; 0,013; y 0,027, respectivamente. En los tres sets de datos la propuesta evaluada en esta investigación es mejor que CNN.

Finalmente, como se puede observar en los cuadros 5.5 y 5.6, la mejora del rendimiento en términos de macro- F_1 del 40% (con SVM) y 50% (con ANN) promediado usando la representación *bin-rfl* en lugar de *bin-idf* y la transformación Relevancia Binaria.

Acá se puede mencionar también que [40] se reportan medidas de desempeño utilizando los algunos sets de datos utilizado en esta investigación: Science, Education, Enron y Bibtex, comparando su propuesta PNML, logrando un macro- F_1 de 0,298; 0,31; 0,262 y 0,418, respectivamente. En esta comparación, nuestra propuesta logra un macro- F_1 de 0,461; 0,285; 0,319 y 0,423, respectivamente. A partir de lo anterior, es posible apreciar que los modelos de aprendizaje superficial pueden entregar mejores resultados que algunos modelos de aprendizaje profundo en tres de los diez set de datos comparados.

TP	Repr.	Reuters	Ohsumed	Enron	Slashdot	Langlog
BR	<i>bin-idf</i>	0,230	0,379	0,223	0,312	0,075
BR	<i>bin-rfl</i>	0,239	0,426	0,230	0,377	0,279
LP	<i>bin-idf</i>	0,236	0,361	0,182	0,336	0,062
LP	<i>bin-rfl</i>	0,222	0,420	0,220	0,346	0,179
TP	Repr.	Bibtex	Medical	TMC2007	Science	Education
BR	<i>bin-idf</i>	0,322	0,379	0,492	0,286	0,236
BR	<i>bin-rfl</i>	0,351	0,345	0,527	0,293	0,210
LP	<i>bin-idf</i>	0,272	0,359	0,278	0,222	0,140
LP	<i>bin-rfl</i>	0,300	0,367	0,336	0,325	0,226

CUADRO 5.5. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con **SVM** en términos de **macro- F_1**

TP	Repr.	Reuters	Ohsumed	Enron	Slashdot	Langlog
BR	<i>bin-idf</i>	0,240	0,350	0,229	0,337	0,081
BR	<i>bin-rfl</i>	0,297	0,411	0,319	0,439	0,307
LP	<i>bin-idf</i>	0,220	0,349	0,166	0,284	0,052
LP	<i>bin-rfl</i>	0,268	0,402	0,199	0,364	0,126
TP	Repr.	Bibtex	Medical	TMC2007	Science	Education
BR	<i>bin-idf</i>	0,325	0,381	0,527	0,383	0,304
BR	<i>bin-rfl</i>	0,423	0,408	0,544	0,461	0,284
LP	<i>bin-idf</i>	0,167	0,328	0,370	0,215	0,198
LP	<i>bin-rfl</i>	0,222	0,398	0,453	0,313	0,285

CUADRO 5.6. Resultados experimentales de diferentes transformaciones del problema (TP: BR y LP) y Representaciones con ANN en términos de **macro- F_1**

Para presentar el impacto de la función *rfl* en los resultados experimentales, la Figura 5.1 muestra gráficamente como, en casi todos los casos, la representación *bin-rfl* presenta mejoras significativas en relación a *bin-idf*. Este porcentaje se calcula como la razón entre la diferencia de la métrica con la nueva representación y la antigua representación. A partir de ella, se puede observar que las mejoras, en muchos casos son superiores al 20 %, en términos de macro- F_1 .

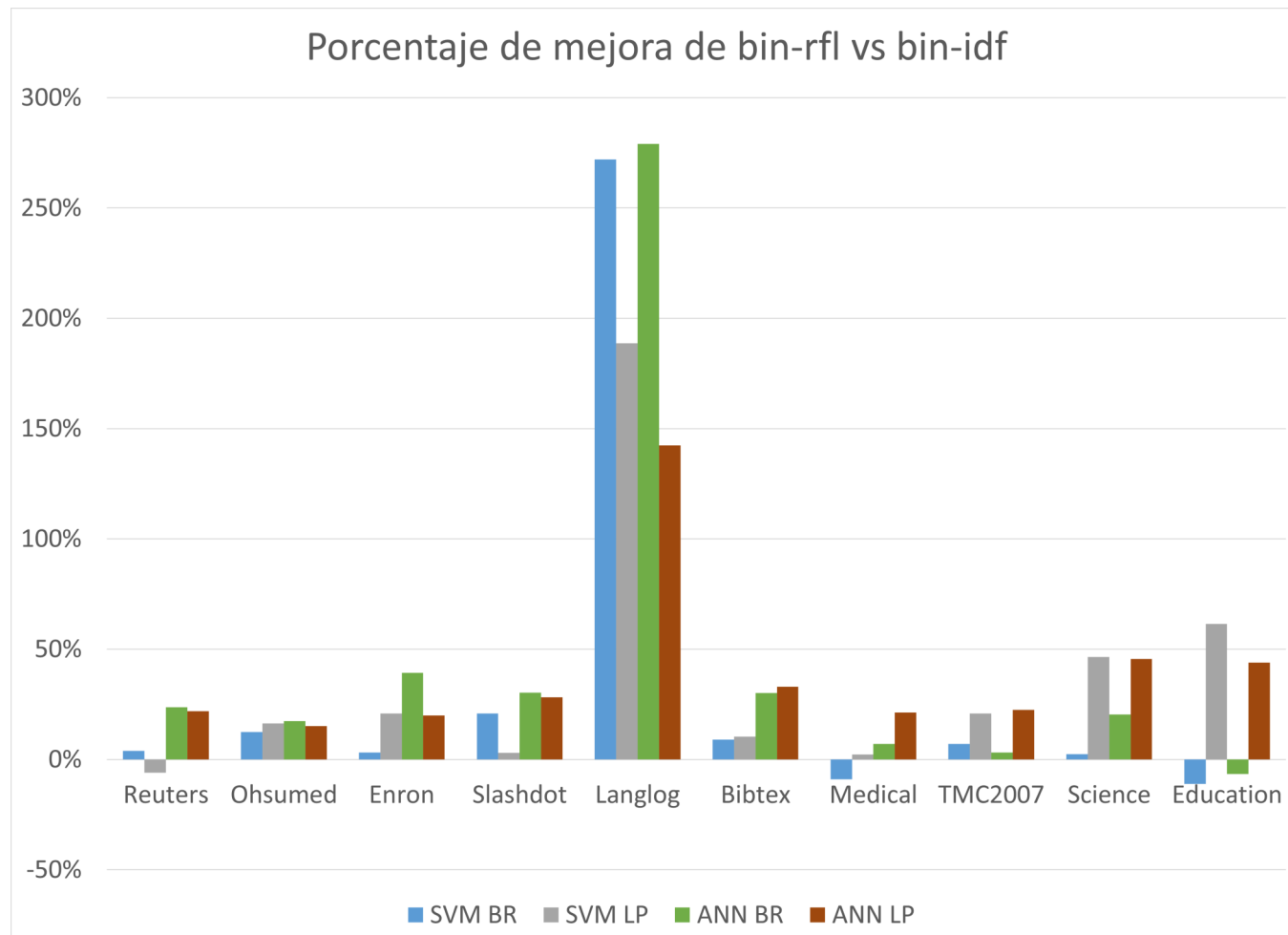


FIGURA 5.1. Porcentaje de mejora del desempeño en términos de macro- F_1

Con el objetivo de analizar la relación de las mejoras en los desempeños introducidos por la función rfl en las diferentes métricas evaluadas: *Hamming Loss*, *Accuracy* y macro- F_1 , con las diferentes características de los set de documentos analizados: Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad, se realizó un análisis de correlación de las métricas y las características, identificando relaciones que se explican a continuación.

Es necesario recordar que la métrica de Cardinalidad se calcula como el número de etiquetas promedio que tiene un documento, la Densidad como la cardinalidad dividida por la cantidad total de etiquetas y la Diversidad como el porcentaje de conjuntos de etiquetas presentes en el conjunto de documentos dividido por el número de conjuntos de etiquetas posibles.

En primer lugar, se analiza la relación en la métrica *Hamming Loss*, tal como muestra la Figura 5.2 . En este análisis es posible identificar correlación inversa entre el uso de SVM con la transformación del problema *Label Powerset* con el número de etiquetas y con la diversidad de etiquetas. Asimismo, una correlación directa entre esta transformación del problema y el número de documentos, tamaño del vocabulario densidad de etiquetas y diversidad de etiquetas. Asimismo, es posible apreciar que existe una correlación inversa entre el uso de ANN con la transformación del problema *Label Powerset* con la cardinalidad y la diversidad del set de documentos.

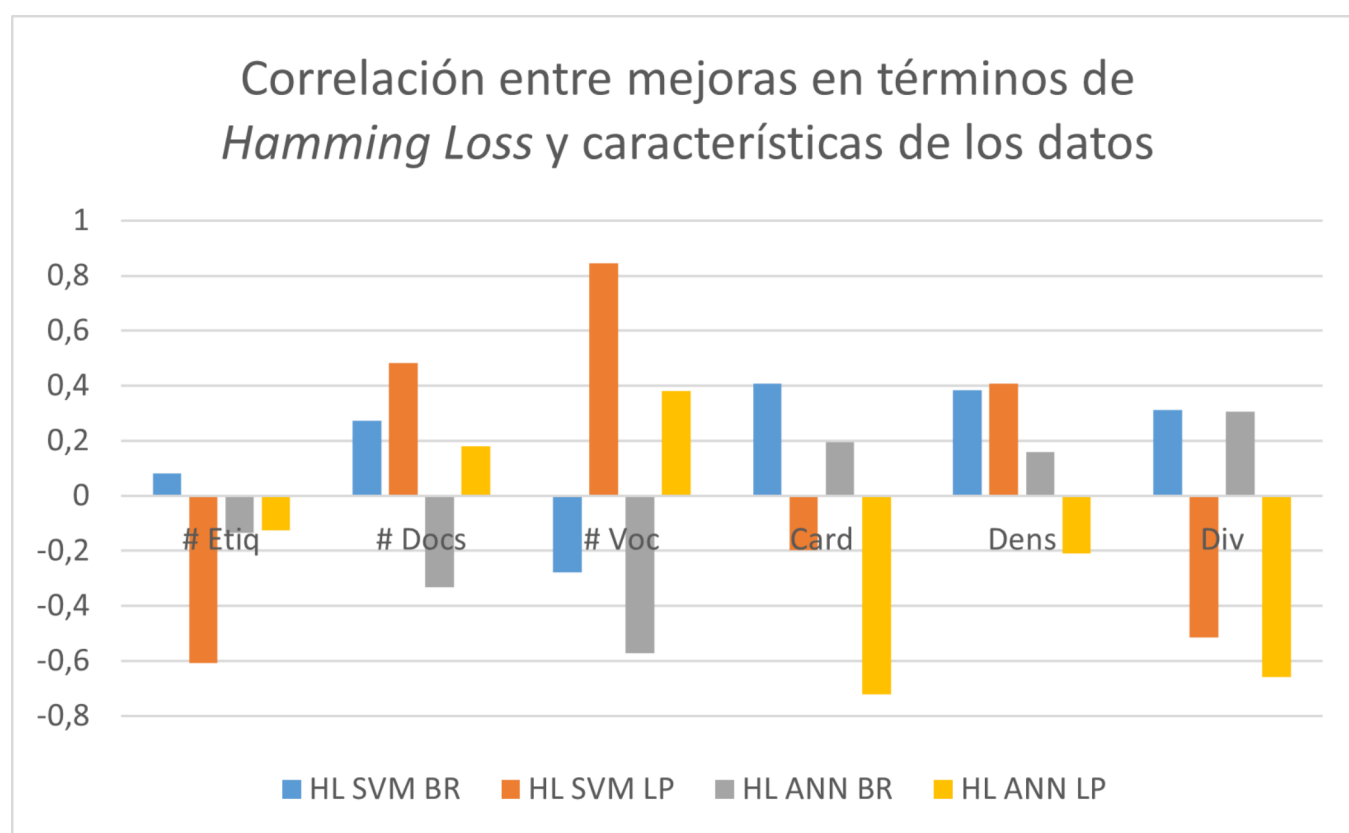


FIGURA 5.2. Correlación entre las mejoras del desempeño en términos de *Hamming Loss* y las diferentes características de los set de datos (Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad)

En segundo lugar, se analiza la relación en la métrica *Accuracy*, la cual, tal como se muestra en la Figura 5.3, presenta una correlación inversa con el número de documentos y una correlación directa con la diversidad de etiquetas. Asimismo, se puede ver que el uso de SVM con la transformación *Label Powerset* obtiene mejores desempeños con un menor número de documentos, menor tamaño del vocabulario y menor

valor de las medidas de cardinalidad y densidad de etiquetas.

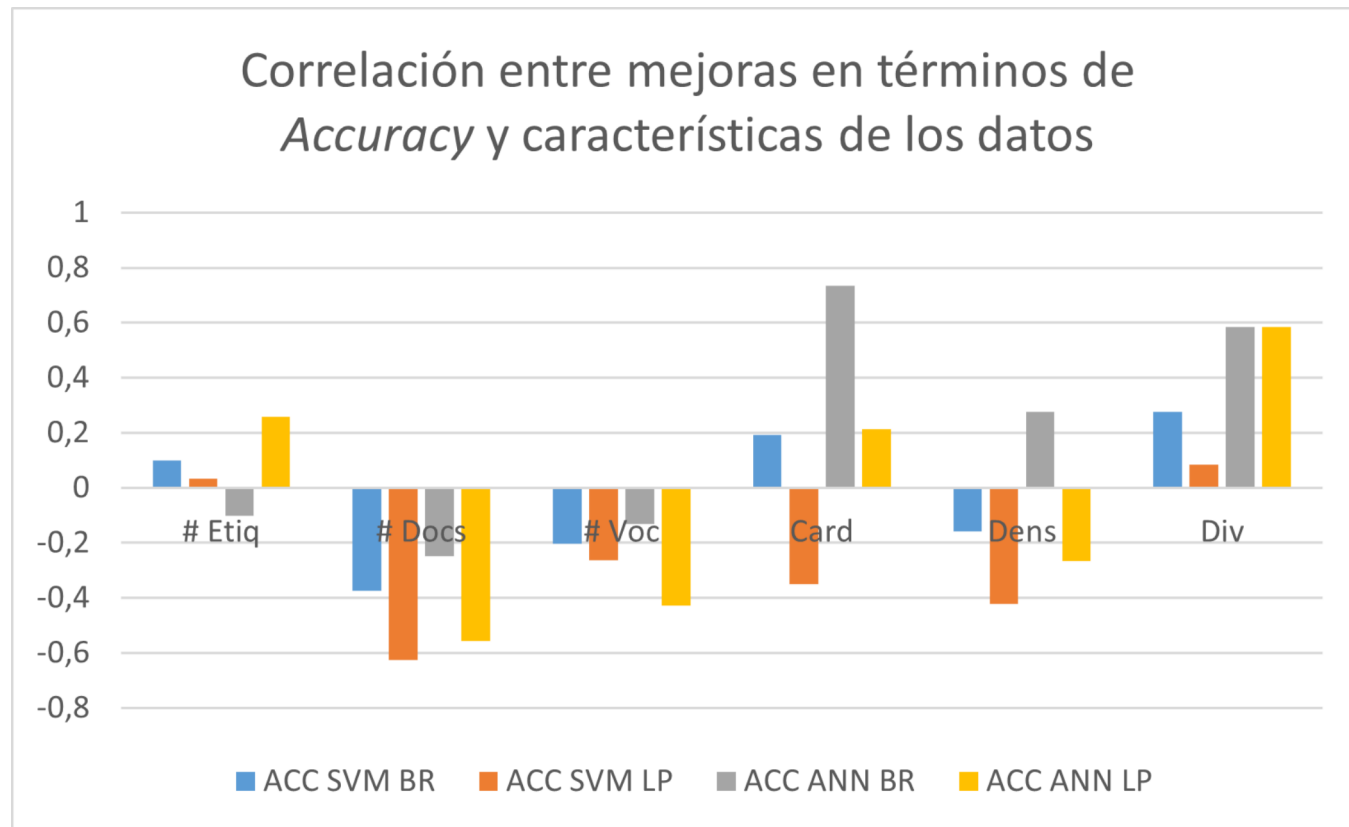


FIGURA 5.3. Correlación entre las mejoras del desempeño en términos de *Accuracy* y las diferentes características de los set de datos (Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad)

En tercer lugar, tal como se muestra en la Figura 5.4, se analiza la relación en la métrica macro- F_1 con los diferentes conjuntos de documentos. En esta medida de desempeño es posible identificar correlación negativa de los dos clasificadores (SVM y ANN) con las dos transformaciones del problema (BT y LP) con el número de etiquetas, el número de documentos y el tamaño del vocabulario. Asimismo, se presenta una correlación directa con la cardinalidad, densidad y diversidad de etiquetas. Esto se puede interpretar, como que a menor número de documentos o menor tamaño del vocabulario, la mejora que introduce la función *rfl* es mayor. Y también, que a mayor cardinalidad de etiquetas, diversidad de etiquetas y, en menor medida, a la densidad de etiquetas, la mejora que introduce la función *rfl* es mayor en la medida macro- F_1 .

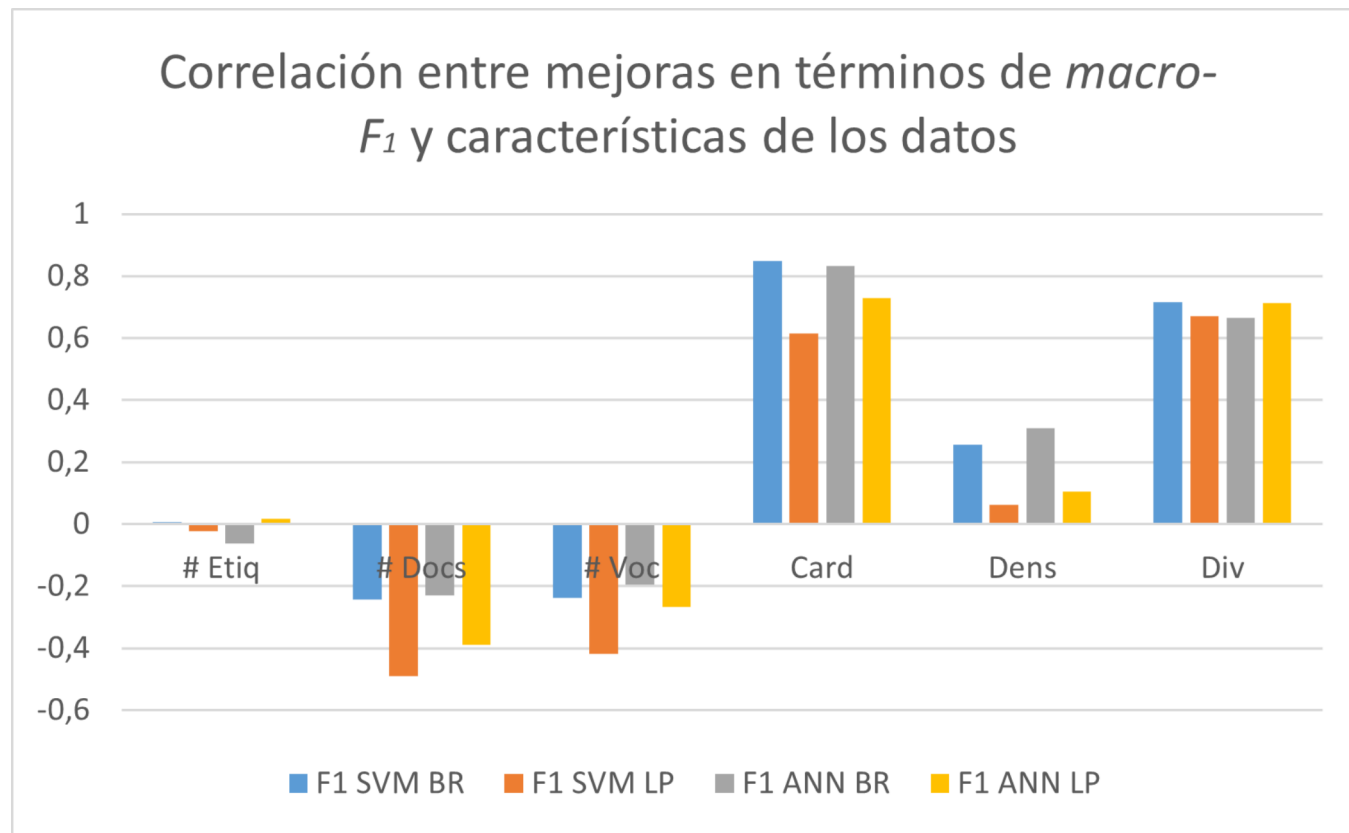


FIGURA 5.4. Correlación entre las mejoras del desempeño en términos de $macro-F_1$ y las diferentes características de los set de datos (Número de etiquetas, Número de documentos, Número de términos en el vocabulario, Cardinalidad, Densidad y Diversidad)

Para evaluar los resultados como en [1], se implementó una prueba basada en *two-tailed paired t-test* al nivel de significación del 5%. De acuerdo con estos resultados, la transformación del problema de Relevancia Binaria, con ANN y *bin-rfl* es mejor que la Relevancia Binaria con ANN y *bin-idf* en todas las medidas ($p = 0,0103$ para *Accuracy*, $p = 0,0491$ para *Hamming Loss* y $p = 0,0078$ para F_1). El valor de p mostrado entre paréntesis proporciona una cuantificación adicional del nivel de significación.

Conclusiones y Trabajo Futuro

6.1. Conclusiones

El crecimiento del interés en modelos de aprendizaje profundo no descarta las técnicas y modelos basados en aprendizaje superficial, sobretudo cuando el conjunto de casos de entrenamiento es de menor volumen y el conjunto de características no es muy extenso. Las aproximaciones de “caja blanca” versus las de “caja negra”, cuentan con algunas ventajas, donde se destaca la factibilidad de trabajar con conjuntos relativamente pequeños de datos y la interpretabilidad de los resultados. Cuestiones que en algunos campos de aplicación son fundamentales.

La clasificación automática con múltiples etiquetas es un tema importante en la recuperación de la información y el aprendizaje automático que ha tomado mayor relevancia en los últimos años. La representación y clasificación de textos se han tratado tradicionalmente usando $tf - idf$ debido a su simplicidad y buen desempeño. Sin embargo, la representación $tf - idf$ no toma en cuenta que los ejemplos pueden tener distintas etiquetas. Esto último toma mucha relevancia en conjuntos de datos con una alta cardinalidad y diversidad de etiquetas.

Los cambios en la representación de entrada a los clasificadores automáticos pueden utilizar los conocimientos sobre el problema, su dominio, una etiqueta en particular o la categoría a la que pertenece el documento. La función rfl puede ser desarrollada para resolver un problema particular directamente, sin complejas transformaciones de problemas, utilizando la información de los ejemplos y sus diferentes etiquetas.

En este trabajo, hemos presentado la función rfl para construir nuevas representaciones de texto para el enfoque de clasificación multi-etiqueta. Esta función permite discriminar los términos que mejor describen una categoría, en contraste con otras categorías, aprovechando así las características del dominio de documentos que conforman el corpus.

Esta propuesta se evaluó utilizando dos diferentes clasificadores lineales, las Redes Neuronales Artificiales (ANN) y las Máquinas de Soporte Vectorial (SVM), con el objetivo de evaluar el impacto de la función en clasificadores simples. A su vez, se evaluó el impacto en diez conjuntos de textos diferentes, que corresponden a artículos científicos médicos, documentos periodísticos, informes de diagnósticos médicos, mensajes de correo electrónico y páginas web. Se realiza una comparación con $bin - idf$ y se utilizan dos transformaciones del problema multi-etiquetado (Relevancia Binaria y *Label Powerset*).

El desempeño de esta función muestra una mejora en casi todos los casos, utilizando la transformación Relevancia Binaria y Máquinas de Soporte Vectorial. Solo en la medida *Hamming Loss*, fue mejor usar *Label Powerset* y Máquinas de Soporte Vectorial.

El mayor impacto de la utilización de la función rfl se produce en la métrica de desempeño macro- F_1 cuando los conjuntos de datos tienen menor número de etiquetas, menor número de documentos y menor tamaño del vocabulario. Asimismo, esta medida mejora en conjuntos de datos con mayor cardinalidad, densidad y diversidad de etiquetas. Esto refleja la utilidad de la función en conjuntos de datos de menor volumen.

Creemos que la contribución de la utilización de la función rfl , al utilizarla como factor de ponderación para modificar la representación multi-etiqueta, se debe a una mejor resolución del problema considerado, ya que es capaz de hacer una mejor identificación de los términos en los documentos, lo que se refleja en un mejor rendimiento de los modelos de clasificación automática.

6.2. Trabajo Futuro

En futuros estudios, planeamos utilizar la función rfl para la tarea de selección de características o identificación de atributos más significativos para discriminar. Además se podrían utilizar otras representaciones, por ejemplo *Part of Speech*, N-gramas o basadas en otras distribuciones de probabilidad para construir una representación dependiente de etiqueta.

Asimismo, analizaremos en profundidad el impacto de la función rfl en el desempeño de clasificadores no lineales, tales como *Random Forest* y *Decision Tree*. Resultados previos muestran mejoras importantes en estos clasificadores no lineales, y el desafío es comprender cómo estos clasificadores reconocen los cambios que provoca la función rfl para mejorar su desempeño.

También utilizaremos la función rfl para procesar las salidas de modelos de aprendizaje más complejos, por ejemplo con *word2vec*, de modo de mejorar su desempeño, a partir de la incorporación de información de las etiquetas para ponderar los conceptos sintetizados.

Otra línea de trabajo es incorporar a la función rfl ponderadores que permitan atacar el problema del desbalance, muy común en clasificación multi-etiqueta. Esto se puede lograr agregando como parámetro de la función rfl el número de documentos por cada etiqueta en relación al número total de documentos y de etiquetas.

Finalmente, usaremos la representación para realizar análisis de opiniones, clasificación de correo electrónico y otras aplicaciones de reconocimiento de patrones.

Agradecimientos

Agradezco a Conicyt por la beca 21080149, a la Universidad Técnica Federico Santa María por el apoyo y a la Pontificia Universidad Católica de Valparaíso. A mi familia, mis amigos y mis profesores.

Anexos

6.3. Modelo Bernoulli Multivariado

La distribución de Bernoulli es utilizada para modelar las variables aleatorias resultantes de un experimento binario (considerando $Rec(X) = \{0, 1\}$) mediante un único parámetro p , el cual se interpreta como la probabilidad de obtener un éxito en dicho experimento. La función de cuantía es la siguiente:

$$(6.1) \quad P(X = x) = p^x \cdot (1 - p)^{1-x} \cdot \mathbf{1}_{\{0,1\}}(x),$$

donde la función indicatriz de $X: \mathbf{1}_A : X \rightarrow \{0, 1\}$ es definida como:

$$(6.2) \quad \mathbf{1}_A(x) = \begin{cases} 1 & \text{si el experimento es exitoso,} \\ 0 & \text{si el experimento no es exitoso.} \end{cases}$$

La variable aleatoria definida de esta manera tiene esperanza p y varianza $p(1 - p)$.

Para el caso Bernoulli Multivariado se tiene un conjunto de variables Bernoulli. En este caso la función de cuantía es la siguiente:

$$(6.3) \quad P(X_1 = x_1, \dots, X_n = x_n) = p_1^{x_1} (1 - p_1)^{1-x_1} \dots p_n^{x_n} (1 - p_n)^{1-x_n}.$$

6.4. Modelo Multinomial

En teoría de probabilidad, la distribución multinomial es una generalización de la distribución binomial.

La distribución binomial es la probabilidad de un número de éxitos en N sucesos de Bernoulli independientes, con la misma probabilidad de éxito en cada suceso. En una distribución multinomial, el análogo a la distribución de Bernoulli es la distribución categórica, donde cada suceso concluye en únicamente un resultado de un número finito K de los posibles, con probabilidades p_1, \dots, p_k , tal que $p_i \geq 0$ para i entre 1 y K y $\sum_{i=1}^k p_i = 1$; y con n sucesos independientes.

Entonces sea la variable aleatoria X_i , que indica el número de veces que se ha dado el resultado i sobre los n sucesos. El vector $X = (X_1, \dots, X_k)$ sigue una distribución multinomial con parámetros n y p , donde $p = (p_1, \dots, p_k)$.

La función de probabilidad de la distribución multinomial es:

$$(6.4) \quad f(x_1, \dots, x_k; n, p_1, \dots, p_k) = \Pr(X_1 = x_1 \text{ y } \dots \text{ y } X_k = x_k)$$

$$= \begin{cases} \frac{n!}{x_1! \cdots x_k!} p_1^{x_1} \cdots p_k^{x_k}, & \text{cuando } \sum_{i=1}^k x_i = n \\ 0 & \text{En otros casos.} \end{cases}$$

6.5. Tablas con más resultados experimentales

6.5.1. Resultados de Reuters. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,273 \pm 0,012$	$0,161 \pm 0,005$	$0,295 \pm 0,016$	$0,123 \pm 0,005$	$0,012 \pm 0,0$
Decision Tree	$0,234 \pm 0,008$	$0,212 \pm 0,008$	$0,205 \pm 0,011$	$0,241 \pm 0,01$	$0,019 \pm 0,001$
Naive Bayes	$0,02 \pm 0,003$	$0,053 \pm 0,004$	$0,033 \pm 0,005$	$0,39 \pm 0,018$	$0,227 \pm 0,009$
Linear SVM	$0,247 \pm 0,007$	$0,23 \pm 0,013$	$0,23 \pm 0,011$	$0,263 \pm 0,022$	$0,017 \pm 0,0$
Neural Net	$0,298 \pm 0,006$	$0,24 \pm 0,015$	$0,266 \pm 0,017$	$0,237 \pm 0,021$	$0,015 \pm 0,0$
RBF SVM	$0,201 \pm 0,008$	$0,11 \pm 0,005$	$0,258 \pm 0,01$	$0,077 \pm 0,004$	$0,012 \pm 0,0$
Adaboost	$0,263 \pm 0,006$	$0,252 \pm 0,011$	$0,325 \pm 0,021$	$0,225 \pm 0,013$	$0,013 \pm 0,0$
Random Forest	$0,304 \pm 0,006$	$0,166 \pm 0,012$	$0,299 \pm 0,025$	$0,126 \pm 0,008$	$0,012 \pm 0,0$

CUADRO 6.1. Resultados de Reuters con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,332 \pm 0,014$	$0,198 \pm 0,015$	$0,348 \pm 0,032$	$0,155 \pm 0,011$	$0,011 \pm 0,0$
Decision Tree	$0,646 \pm 0,006$	$0,429 \pm 0,02$	$0,433 \pm 0,021$	$0,447 \pm 0,026$	$0,009 \pm 0,0$
Naive Bayes	$0,026 \pm 0,005$	$0,047 \pm 0,002$	$0,027 \pm 0,004$	$0,394 \pm 0,015$	$0,223 \pm 0,001$
Linear SVM	$0,2 \pm 0,015$	$0,229 \pm 0,016$	$0,221 \pm 0,015$	$0,271 \pm 0,025$	$0,018 \pm 0,001$
Neural Net	$0,373 \pm 0,014$	$0,297 \pm 0,018$	$0,314 \pm 0,022$	$0,298 \pm 0,017$	$0,013 \pm 0,0$
RBF SVM	$0,126 \pm 0,011$	$0,089 \pm 0,01$	$0,276 \pm 0,022$	$0,062 \pm 0,008$	$0,013 \pm 0,0$
Adaboost	$0,55 \pm 0,011$	$0,411 \pm 0,013$	$0,501 \pm 0,012$	$0,371 \pm 0,016$	$0,008 \pm 0,0$
Random Forest	$0,599 \pm 0,006$	$0,332 \pm 0,012$	$0,483 \pm 0,023$	$0,275 \pm 0,01$	$0,007 \pm 0,0$

CUADRO 6.2. Resultados de Reuters con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,281 \pm 0,012$	$0,175 \pm 0,005$	$0,337 \pm 0,019$	$0,13 \pm 0,004$	$0,012 \pm 0,0$
Decision Tree	$0,233 \pm 0,022$	$0,254 \pm 0,019$	$0,272 \pm 0,015$	$0,257 \pm 0,023$	$0,016 \pm 0,0$
Naive Bayes	$0,022 \pm 0,006$	$0,044 \pm 0,001$	$0,024 \pm 0,001$	$0,386 \pm 0,021$	$0,232 \pm 0,006$
Linear SVM	$0,267 \pm 0,017$	$0,246 \pm 0,004$	$0,248 \pm 0,009$	$0,277 \pm 0,006$	$0,016 \pm 0,0$
Neural Net	$0,317 \pm 0,015$	$0,265 \pm 0,004$	$0,291 \pm 0,005$	$0,262 \pm 0,007$	$0,015 \pm 0,001$
RBF SVM	$0,231 \pm 0,011$	$0,137 \pm 0,005$	$0,298 \pm 0,015$	$0,097 \pm 0,004$	$0,012 \pm 0,0$
Adaboost	$0,294 \pm 0,014$	$0,263 \pm 0,012$	$0,343 \pm 0,019$	$0,229 \pm 0,011$	$0,012 \pm 0,0$
Random Forest	$0,29 \pm 0,012$	$0,16 \pm 0,013$	$0,333 \pm 0,022$	$0,115 \pm 0,009$	$0,011 \pm 0,0$

CUADRO 6.3. Resultados de Reuters con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,29 \pm 0,015$	$0,184 \pm 0,017$	$0,334 \pm 0,024$	$0,142 \pm 0,014$	$0,012 \pm 0,0$
Decision Tree	$0,299 \pm 0,011$	$0,311 \pm 0,018$	$0,317 \pm 0,018$	$0,327 \pm 0,023$	$0,014 \pm 0,001$
Naive Bayes	$0,026 \pm 0,003$	$0,049 \pm 0,001$	$0,027 \pm 0,001$	$0,386 \pm 0,014$	$0,214 \pm 0,009$
Linear SVM	$0,207 \pm 0,015$	$0,247 \pm 0,013$	$0,232 \pm 0,013$	$0,305 \pm 0,014$	$0,018 \pm 0,0$
Neural Net	$0,36 \pm 0,017$	$0,311 \pm 0,012$	$0,33 \pm 0,01$	$0,315 \pm 0,02$	$0,013 \pm 0,0$
RBF SVM	$0,132 \pm 0,009$	$0,095 \pm 0,008$	$0,294 \pm 0,023$	$0,066 \pm 0,007$	$0,013 \pm 0,0$
Adaboost	$0,344 \pm 0,015$	$0,327 \pm 0,015$	$0,403 \pm 0,02$	$0,296 \pm 0,014$	$0,011 \pm 0,0$
Random Forest	$0,338 \pm 0,019$	$0,222 \pm 0,014$	$0,381 \pm 0,012$	$0,172 \pm 0,014$	$0,01 \pm 0,0$

CUADRO 6.4. Resultados de Reuters con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,335 \pm 0,022$	$0,17 \pm 0,006$	$0,249 \pm 0,019$	$0,158 \pm 0,005$	$0,016 \pm 0,0$
Decision Tree	$0,356 \pm 0,02$	$0,181 \pm 0,015$	$0,205 \pm 0,018$	$0,177 \pm 0,014$	$0,017 \pm 0,0$
Naive Bayes	$0,104 \pm 0,008$	$0,12 \pm 0,01$	$0,133 \pm 0,007$	$0,132 \pm 0,015$	$0,023 \pm 0,0$
Linear SVM	$0,353 \pm 0,013$	$0,236 \pm 0,012$	$0,237 \pm 0,015$	$0,263 \pm 0,019$	$0,017 \pm 0,0$
Neural Net	$0,358 \pm 0,007$	$0,22 \pm 0,014$	$0,245 \pm 0,018$	$0,216 \pm 0,017$	$0,016 \pm 0,0$
RBF SVM	$0,38 \pm 0,011$	$0,146 \pm 0,013$	$0,243 \pm 0,019$	$0,133 \pm 0,011$	$0,015 \pm 0,0$
Adaboost	$0,091 \pm 0,006$	$0,01 \pm 0,0$	$0,009 \pm 0,0$	$0,018 \pm 0,0$	$0,022 \pm 0,0$
Random Forest	$0,392 \pm 0,014$	$0,195 \pm 0,014$	$0,246 \pm 0,019$	$0,182 \pm 0,011$	$0,015 \pm 0,0$

CUADRO 6.5. Resultados de Reuters con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,414 \pm 0,012$	$0,225 \pm 0,005$	$0,267 \pm 0,012$	$0,223 \pm 0,007$	$0,016 \pm 0,0$
Decision Tree	$0,734 \pm 0,012$	$0,389 \pm 0,017$	$0,411 \pm 0,021$	$0,391 \pm 0,014$	$0,01 \pm 0,0$
Naive Bayes	$0,141 \pm 0,008$	$0,135 \pm 0,011$	$0,156 \pm 0,014$	$0,143 \pm 0,011$	$0,022 \pm 0,0$
Linear SVM	$0,35 \pm 0,013$	$0,222 \pm 0,013$	$0,219 \pm 0,016$	$0,257 \pm 0,009$	$0,019 \pm 0,001$
Neural Net	$0,462 \pm 0,012$	$0,268 \pm 0,018$	$0,293 \pm 0,028$	$0,263 \pm 0,015$	$0,014 \pm 0,0$
RBF SVM	$0,418 \pm 0,019$	$0,158 \pm 0,015$	$0,286 \pm 0,024$	$0,14 \pm 0,014$	$0,014 \pm 0,0$
Adaboost	$0,104 \pm 0,005$	$0,019 \pm 0,005$	$0,03 \pm 0,014$	$0,025 \pm 0,005$	$0,022 \pm 0,0$
Random Forest	$0,748 \pm 0,01$	$0,409 \pm 0,027$	$0,432 \pm 0,03$	$0,411 \pm 0,028$	$0,009 \pm 0,001$

CUADRO 6.6. Resultados de Reuters con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,348 \pm 0,016$	$0,187 \pm 0,023$	$0,266 \pm 0,032$	$0,168 \pm 0,02$	$0,016 \pm 0,0$
Decision Tree	$0,358 \pm 0,015$	$0,197 \pm 0,008$	$0,22 \pm 0,011$	$0,191 \pm 0,01$	$0,016 \pm 0,0$
Naive Bayes	$0,111 \pm 0,006$	$0,131 \pm 0,008$	$0,139 \pm 0,01$	$0,145 \pm 0,011$	$0,023 \pm 0,0$
Linear SVM	$0,368 \pm 0,014$	$0,262 \pm 0,015$	$0,274 \pm 0,013$	$0,27 \pm 0,018$	$0,017 \pm 0,001$
Neural Net	$0,375 \pm 0,012$	$0,23 \pm 0,016$	$0,253 \pm 0,021$	$0,227 \pm 0,015$	$0,016 \pm 0,001$
RBF SVM	$0,393 \pm 0,017$	$0,159 \pm 0,01$	$0,271 \pm 0,015$	$0,142 \pm 0,008$	$0,014 \pm 0,0$
Adaboost	$0,094 \pm 0,008$	$0,007 \pm 0,001$	$0,011 \pm 0,008$	$0,015 \pm 0,001$	$0,022 \pm 0,0$
Random Forest	$0,412 \pm 0,01$	$0,203 \pm 0,015$	$0,268 \pm 0,014$	$0,188 \pm 0,016$	$0,014 \pm 0,0$

CUADRO 6.7. Resultados de Reuters con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,373 \pm 0,012$	$0,208 \pm 0,005$	$0,261 \pm 0,013$	$0,199 \pm 0,005$	$0,016 \pm 0,0$
Decision Tree	$0,429 \pm 0,016$	$0,241 \pm 0,013$	$0,257 \pm 0,017$	$0,24 \pm 0,015$	$0,015 \pm 0,0$
Naive Bayes	$0,162 \pm 0,009$	$0,159 \pm 0,012$	$0,173 \pm 0,011$	$0,174 \pm 0,021$	$0,021 \pm 0,0$
Linear SVM	$0,356 \pm 0,018$	$0,249 \pm 0,004$	$0,248 \pm 0,009$	$0,277 \pm 0,009$	$0,018 \pm 0,001$
Neural Net	$0,44 \pm 0,01$	$0,28 \pm 0,01$	$0,305 \pm 0,012$	$0,277 \pm 0,013$	$0,014 \pm 0,0$
RBF SVM	$0,418 \pm 0,013$	$0,172 \pm 0,008$	$0,31 \pm 0,013$	$0,15 \pm 0,008$	$0,014 \pm 0,0$
Adaboost	$0,103 \pm 0,011$	$0,009 \pm 0,002$	$0,01 \pm 0,004$	$0,016 \pm 0,001$	$0,022 \pm 0,0$
Random Forest	$0,475 \pm 0,006$	$0,269 \pm 0,02$	$0,314 \pm 0,025$	$0,26 \pm 0,019$	$0,014 \pm 0,0$

CUADRO 6.8. Resultados de Reuters con Label Powerset- tf-rfl

6.5.2. Resultados de Enron. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,083 \pm 0,016$	$0,092 \pm 0,012$	$0,186 \pm 0,031$	$0,074 \pm 0,013$	$0,06 \pm 0,001$
Decision Tree	$0,081 \pm 0,012$	$0,2 \pm 0,012$	$0,198 \pm 0,014$	$0,218 \pm 0,009$	$0,07 \pm 0,002$
Naive Bayes	$0,0 \pm 0,0$	$0,154 \pm 0,006$	$0,127 \pm 0,008$	$0,346 \pm 0,018$	$0,203 \pm 0,016$
Linear SVM	$0,08 \pm 0,006$	$0,223 \pm 0,01$	$0,237 \pm 0,008$	$0,227 \pm 0,015$	$0,065 \pm 0,001$
Neural Net	$0,097 \pm 0,014$	$0,229 \pm 0,018$	$0,261 \pm 0,021$	$0,219 \pm 0,02$	$0,057 \pm 0,002$
RBF SVM	$0,109 \pm 0,016$	$0,104 \pm 0,008$	$0,183 \pm 0,01$	$0,089 \pm 0,007$	$0,046 \pm 0,001$
Adaboost	$0,098 \pm 0,014$	$0,218 \pm 0,014$	$0,281 \pm 0,031$	$0,196 \pm 0,013$	$0,052 \pm 0,001$
Random Forest	$0,112 \pm 0,005$	$0,177 \pm 0,004$	$0,286 \pm 0,018$	$0,145 \pm 0,009$	$0,049 \pm 0,001$

CUADRO 6.9. Resultados de Enron con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,12 \pm 0,012$	$0,139 \pm 0,007$	$0,236 \pm 0,021$	$0,11 \pm 0,005$	$0,047 \pm 0,001$
Decision Tree	$0,183 \pm 0,019$	$0,224 \pm 0,017$	$0,221 \pm 0,02$	$0,25 \pm 0,019$	$0,057 \pm 0,001$
Naive Bayes	$0,004 \pm 0,002$	$0,133 \pm 0,012$	$0,149 \pm 0,014$	$0,308 \pm 0,027$	$0,206 \pm 0,012$
Linear SVM	$0,105 \pm 0,02$	$0,23 \pm 0,018$	$0,274 \pm 0,018$	$0,221 \pm 0,022$	$0,05 \pm 0,001$
Neural Net	$0,225 \pm 0,014$	$0,319 \pm 0,019$	$0,353 \pm 0,019$	$0,313 \pm 0,022$	$0,039 \pm 0,001$
RBF SVM	$0,002 \pm 0,001$	$0,046 \pm 0,008$	$0,096 \pm 0,012$	$0,034 \pm 0,009$	$0,055 \pm 0,002$
Adaboost	$0,185 \pm 0,014$	$0,227 \pm 0,017$	$0,292 \pm 0,023$	$0,199 \pm 0,017$	$0,042 \pm 0,001$
Random Forest	$0,207 \pm 0,01$	$0,178 \pm 0,014$	$0,294 \pm 0,039$	$0,145 \pm 0,011$	$0,038 \pm 0,001$

CUADRO 6.10. Resultados de Enron con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,1 \pm 0,016$	$0,098 \pm 0,009$	$0,168 \pm 0,024$	$0,089 \pm 0,01$	$0,071 \pm 0,002$
Decision Tree	$0,115 \pm 0,015$	$0,158 \pm 0,015$	$0,173 \pm 0,019$	$0,157 \pm 0,017$	$0,069 \pm 0,002$
Naive Bayes	$0,096 \pm 0,025$	$0,12 \pm 0,014$	$0,187 \pm 0,022$	$0,118 \pm 0,013$	$0,067 \pm 0,004$
Linear SVM	$0,163 \pm 0,013$	$0,182 \pm 0,024$	$0,226 \pm 0,036$	$0,168 \pm 0,022$	$0,056 \pm 0,003$
Neural Net	$0,122 \pm 0,009$	$0,166 \pm 0,018$	$0,173 \pm 0,02$	$0,173 \pm 0,02$	$0,069 \pm 0,001$
RBF SVM	$0,165 \pm 0,014$	$0,082 \pm 0,005$	$0,139 \pm 0,026$	$0,088 \pm 0,004$	$0,06 \pm 0,002$
Adaboost	$0,137 \pm 0,017$	$0,061 \pm 0,005$	$0,083 \pm 0,008$	$0,071 \pm 0,005$	$0,064 \pm 0,003$
Random Forest	$0,138 \pm 0,023$	$0,161 \pm 0,015$	$0,188 \pm 0,021$	$0,155 \pm 0,016$	$0,065 \pm 0,001$

CUADRO 6.11. Resultados de Enron con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,113 \pm 0,015$	$0,12 \pm 0,008$	$0,179 \pm 0,014$	$0,111 \pm 0,01$	$0,073 \pm 0,003$
Decision Tree	$0,42 \pm 0,018$	$0,247 \pm 0,03$	$0,267 \pm 0,033$	$0,251 \pm 0,039$	$0,055 \pm 0,002$
Naive Bayes	$0,116 \pm 0,017$	$0,117 \pm 0,016$	$0,211 \pm 0,03$	$0,113 \pm 0,017$	$0,066 \pm 0,002$
Linear SVM	$0,212 \pm 0,01$	$0,22 \pm 0,017$	$0,264 \pm 0,021$	$0,21 \pm 0,026$	$0,057 \pm 0,002$
Neural Net	$0,189 \pm 0,004$	$0,199 \pm 0,028$	$0,203 \pm 0,024$	$0,211 \pm 0,041$	$0,062 \pm 0,002$
RBF SVM	$0,196 \pm 0,012$	$0,086 \pm 0,006$	$0,209 \pm 0,02$	$0,073 \pm 0,004$	$0,062 \pm 0,002$
Adaboost	$0,176 \pm 0,013$	$0,037 \pm 0,005$	$0,137 \pm 0,022$	$0,035 \pm 0,003$	$0,067 \pm 0,003$
Random Forest	$0,421 \pm 0,028$	$0,255 \pm 0,026$	$0,275 \pm 0,025$	$0,261 \pm 0,041$	$0,053 \pm 0,004$

CUADRO 6.12. Resultados de Enron con Label Powerset- bin-rfl

6.5.3. Resultados de Slashdot. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,158 \pm 0,016$	$0,038 \pm 0,003$	$0,256 \pm 0,07$	$0,058 \pm 0,001$	$0,079 \pm 0,001$
Decision Tree	$0,338 \pm 0,011$	$0,342 \pm 0,015$	$0,39 \pm 0,027$	$0,326 \pm 0,023$	$0,046 \pm 0,001$
Naive Bayes	$0,142 \pm 0,009$	$0,163 \pm 0,012$	$0,13 \pm 0,011$	$0,234 \pm 0,015$	$0,109 \pm 0,004$
Linear SVM	$0,257 \pm 0,009$	$0,312 \pm 0,008$	$0,302 \pm 0,009$	$0,334 \pm 0,018$	$0,063 \pm 0,001$
Neural Net	$0,302 \pm 0,015$	$0,337 \pm 0,008$	$0,34 \pm 0,012$	$0,348 \pm 0,022$	$0,055 \pm 0,001$
RBF SVM	$0,222 \pm 0,015$	$0,144 \pm 0,012$	$0,429 \pm 0,045$	$0,099 \pm 0,008$	$0,043 \pm 0,001$
Adaboost	$0,355 \pm 0,016$	$0,347 \pm 0,026$	$0,471 \pm 0,028$	$0,294 \pm 0,03$	$0,041 \pm 0,001$
Random Forest	$0,322 \pm 0,015$	$0,233 \pm 0,016$	$0,4 \pm 0,009$	$0,184 \pm 0,015$	$0,04 \pm 0,001$

CUADRO 6.13. Resultados de Slashdot con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,191 \pm 0,008$	$0,138 \pm 0,016$	$0,573 \pm 0,043$	$0,11 \pm 0,011$	$0,063 \pm 0,001$
Decision Tree	$0,739 \pm 0,016$	$0,586 \pm 0,022$	$0,63 \pm 0,038$	$0,566 \pm 0,027$	$0,018 \pm 0,001$
Naive Bayes	$0,15 \pm 0,005$	$0,168 \pm 0,009$	$0,134 \pm 0,007$	$0,235 \pm 0,013$	$0,106 \pm 0,002$
Linear SVM	$0,325 \pm 0,021$	$0,377 \pm 0,013$	$0,346 \pm 0,014$	$0,436 \pm 0,019$	$0,054 \pm 0,002$
Neural Net	$0,416 \pm 0,012$	$0,439 \pm 0,007$	$0,439 \pm 0,016$	$0,459 \pm 0,012$	$0,042 \pm 0,002$
RBF SVM	$0,241 \pm 0,013$	$0,176 \pm 0,011$	$0,557 \pm 0,026$	$0,113 \pm 0,008$	$0,043 \pm 0,001$
Adaboost	$0,755 \pm 0,016$	$0,645 \pm 0,033$	$0,724 \pm 0,02$	$0,601 \pm 0,037$	$0,018 \pm 0,001$
Random Forest	$0,686 \pm 0,013$	$0,491 \pm 0,012$	$0,653 \pm 0,036$	$0,424 \pm 0,011$	$0,019 \pm 0,001$

CUADRO 6.14. Resultados de Slashdot con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,17 \pm 0,009$	$0,052 \pm 0,011$	$0,262 \pm 0,044$	$0,066 \pm 0,006$	$0,082 \pm 0,002$
Decision Tree	$0,393 \pm 0,023$	$0,305 \pm 0,021$	$0,33 \pm 0,02$	$0,299 \pm 0,025$	$0,055 \pm 0,002$
Naive Bayes	$0,308 \pm 0,017$	$0,195 \pm 0,004$	$0,24 \pm 0,019$	$0,185 \pm 0,004$	$0,064 \pm 0,001$
Linear SVM	$0,425 \pm 0,021$	$0,336 \pm 0,032$	$0,375 \pm 0,035$	$0,318 \pm 0,027$	$0,052 \pm 0,003$
Neural Net	$0,352 \pm 0,024$	$0,284 \pm 0,015$	$0,283 \pm 0,021$	$0,302 \pm 0,016$	$0,062 \pm 0,002$
RBF SVM	$0,436 \pm 0,018$	$0,21 \pm 0,011$	$0,4 \pm 0,046$	$0,204 \pm 0,008$	$0,052 \pm 0,002$
Adaboost	$0,221 \pm 0,011$	$0,041 \pm 0,001$	$0,058 \pm 0,012$	$0,076 \pm 0,002$	$0,076 \pm 0,001$
Random Forest	$0,445 \pm 0,023$	$0,291 \pm 0,011$	$0,33 \pm 0,011$	$0,281 \pm 0,013$	$0,05 \pm 0,002$

CUADRO 6.15. Resultados de Slashdot con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,227 \pm 0,018$	$0,123 \pm 0,016$	$0,343 \pm 0,021$	$0,113 \pm 0,01$	$0,077 \pm 0,002$
Decision Tree	$0,826 \pm 0,004$	$0,591 \pm 0,021$	$0,601 \pm 0,029$	$0,599 \pm 0,025$	$0,022 \pm 0,001$
Naive Bayes	$0,38 \pm 0,019$	$0,232 \pm 0,01$	$0,276 \pm 0,02$	$0,219 \pm 0,006$	$0,059 \pm 0,002$
Linear SVM	$0,442 \pm 0,011$	$0,346 \pm 0,014$	$0,349 \pm 0,017$	$0,358 \pm 0,013$	$0,055 \pm 0,001$
Neural Net	$0,489 \pm 0,014$	$0,364 \pm 0,017$	$0,365 \pm 0,015$	$0,378 \pm 0,018$	$0,051 \pm 0,003$
RBF SVM	$0,526 \pm 0,016$	$0,296 \pm 0,009$	$0,493 \pm 0,01$	$0,262 \pm 0,006$	$0,046 \pm 0,002$
Adaboost	$0,235 \pm 0,015$	$0,05 \pm 0,002$	$0,078 \pm 0,021$	$0,075 \pm 0,002$	$0,076 \pm 0,002$
Random Forest	$0,797 \pm 0,006$	$0,578 \pm 0,036$	$0,622 \pm 0,043$	$0,566 \pm 0,04$	$0,023 \pm 0,001$

CUADRO 6.16. Resultados de Slashdot con Label Powerset- bin-rfl

6.5.4. Resultados de Langlog. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,175 \pm 0,009$	$0,02 \pm 0,003$	$0,066 \pm 0,01$	$0,014 \pm 0,003$	$0,016 \pm 0,0$
Decision Tree	$0,189 \pm 0,01$	$0,078 \pm 0,018$	$0,076 \pm 0,017$	$0,097 \pm 0,023$	$0,028 \pm 0,001$
Naive Bayes	$0,154 \pm 0,01$	$0,034 \pm 0,004$	$0,026 \pm 0,007$	$0,082 \pm 0,005$	$0,055 \pm 0,002$
Linear SVM	$0,24 \pm 0,013$	$0,075 \pm 0,011$	$0,115 \pm 0,022$	$0,063 \pm 0,01$	$0,018 \pm 0,0$
Neural Net	$0,26 \pm 0,022$	$0,081 \pm 0,012$	$0,139 \pm 0,01$	$0,063 \pm 0,012$	$0,016 \pm 0,001$
RBF SVM	$0,156 \pm 0,008$	$0,005 \pm 0,001$	$0,02 \pm 0,007$	$0,003 \pm 0,001$	$0,016 \pm 0,001$
Adaboost	$0,232 \pm 0,006$	$0,076 \pm 0,012$	$0,119 \pm 0,018$	$0,064 \pm 0,012$	$0,018 \pm 0,001$
Random Forest	$0,161 \pm 0,013$	$0,008 \pm 0,004$	$0,034 \pm 0,016$	$0,005 \pm 0,003$	$0,016 \pm 0,001$

CUADRO 6.17. Resultados de Langlog con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,201 \pm 0,017$	$0,067 \pm 0,015$	$0,172 \pm 0,019$	$0,056 \pm 0,013$	$0,016 \pm 0,001$
Decision Tree	$0,562 \pm 0,031$	$0,306 \pm 0,021$	$0,326 \pm 0,025$	$0,32 \pm 0,026$	$0,014 \pm 0,002$
Naive Bayes	$0,156 \pm 0,019$	$0,034 \pm 0,004$	$0,023 \pm 0,002$	$0,092 \pm 0,008$	$0,071 \pm 0,002$
Linear SVM	$0,334 \pm 0,043$	$0,279 \pm 0,025$	$0,317 \pm 0,029$	$0,278 \pm 0,028$	$0,017 \pm 0,001$
Neural Net	$0,403 \pm 0,039$	$0,307 \pm 0,04$	$0,403 \pm 0,045$	$0,273 \pm 0,038$	$0,012 \pm 0,001$
RBF SVM	$0,158 \pm 0,02$	$0,004 \pm 0,001$	$0,013 \pm 0,0$	$0,002 \pm 0,0$	$0,015 \pm 0,001$
Adaboost	$0,695 \pm 0,024$	$0,329 \pm 0,026$	$0,403 \pm 0,041$	$0,296 \pm 0,022$	$0,008 \pm 0,001$
Random Forest	$0,588 \pm 0,03$	$0,184 \pm 0,009$	$0,271 \pm 0,015$	$0,149 \pm 0,01$	$0,01 \pm 0,001$

CUADRO 6.18. Resultados de Langlog con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,221 \pm 0,017$	$0,034 \pm 0,006$	$0,053 \pm 0,013$	$0,036 \pm 0,005$	$0,025 \pm 0,001$
Decision Tree	$0,229 \pm 0,031$	$0,052 \pm 0,011$	$0,054 \pm 0,016$	$0,057 \pm 0,012$	$0,027 \pm 0,001$
Naive Bayes	$0,229 \pm 0,022$	$0,023 \pm 0,004$	$0,031 \pm 0,005$	$0,028 \pm 0,005$	$0,024 \pm 0,001$
Linear SVM	$0,313 \pm 0,022$	$0,062 \pm 0,006$	$0,079 \pm 0,013$	$0,062 \pm 0,004$	$0,02 \pm 0,001$
Neural Net	$0,237 \pm 0,019$	$0,052 \pm 0,008$	$0,055 \pm 0,011$	$0,055 \pm 0,008$	$0,027 \pm 0,001$
RBF SVM	$0,249 \pm 0,037$	$0,013 \pm 0,002$	$0,021 \pm 0,004$	$0,024 \pm 0,003$	$0,023 \pm 0,001$
Adaboost	$0,075 \pm 0,017$	$0,003 \pm 0,001$	$0,002 \pm 0,0$	$0,013 \pm 0,0$	$0,031 \pm 0,003$
Random Forest	$0,239 \pm 0,025$	$0,029 \pm 0,003$	$0,035 \pm 0,006$	$0,034 \pm 0,005$	$0,024 \pm 0,001$

CUADRO 6.19. Resultados de Langlog con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,221 \pm 0,019$	$0,052 \pm 0,014$	$0,077 \pm 0,016$	$0,063 \pm 0,015$	$0,028 \pm 0,002$
Decision Tree	$0,755 \pm 0,029$	$0,297 \pm 0,029$	$0,318 \pm 0,031$	$0,311 \pm 0,023$	$0,013 \pm 0,002$
Naive Bayes	$0,346 \pm 0,032$	$0,053 \pm 0,015$	$0,072 \pm 0,022$	$0,057 \pm 0,015$	$0,021 \pm 0,001$
Linear SVM	$0,519 \pm 0,027$	$0,179 \pm 0,018$	$0,241 \pm 0,023$	$0,176 \pm 0,015$	$0,017 \pm 0,001$
Neural Net	$0,417 \pm 0,015$	$0,126 \pm 0,006$	$0,137 \pm 0,012$	$0,131 \pm 0,012$	$0,022 \pm 0,001$
RBF SVM	$0,355 \pm 0,024$	$0,05 \pm 0,005$	$0,1 \pm 0,009$	$0,055 \pm 0,004$	$0,021 \pm 0,001$
Adaboost	$0,347 \pm 0,05$	$0,069 \pm 0,012$	$0,086 \pm 0,02$	$0,081 \pm 0,017$	$0,024 \pm 0,003$
Random Forest	$0,79 \pm 0,03$	$0,354 \pm 0,042$	$0,388 \pm 0,052$	$0,364 \pm 0,039$	$0,011 \pm 0,002$

CUADRO 6.20. Resultados de Langlog con Label Powerset- bin-rfl

6.5.5. Resultados de Bibtex. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,033 \pm 0,002$	$0,053 \pm 0,004$	$0,142 \pm 0,017$	$0,043 \pm 0,004$	$0,015 \pm 0,0$
Decision Tree	$0,105 \pm 0,006$	$0,291 \pm 0,006$	$0,289 \pm 0,007$	$0,307 \pm 0,007$	$0,02 \pm 0,0$
Naive Bayes	$0,049 \pm 0,001$	$0,097 \pm 0,006$	$0,104 \pm 0,013$	$0,166 \pm 0,011$	$0,05 \pm 0,002$
Linear SVM	$0,122 \pm 0,004$	$0,322 \pm 0,004$	$0,372 \pm 0,006$	$0,313 \pm 0,006$	$0,018 \pm 0,0$
Neural Net	$0,172 \pm 0,004$	$0,325 \pm 0,005$	$0,475 \pm 0,017$	$0,272 \pm 0,006$	$0,013 \pm 0,0$
RBF SVM	$0,115 \pm 0,005$	$0,085 \pm 0,006$	$0,206 \pm 0,017$	$0,069 \pm 0,004$	$0,013 \pm 0,0$
Adaboost	$0,149 \pm 0,008$	$0,317 \pm 0,005$	$0,424 \pm 0,01$	$0,272 \pm 0,005$	$0,014 \pm 0,0$
Random Forest	$0,096 \pm 0,006$	$0,086 \pm 0,006$	$0,285 \pm 0,02$	$0,062 \pm 0,004$	$0,013 \pm 0,0$

CUADRO 6.21. Resultados de Bibtex con con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,107 \pm 0,004$	$0,159 \pm 0,003$	$0,429 \pm 0,027$	$0,115 \pm 0,002$	$0,014 \pm 0,0$
Decision Tree	$0,368 \pm 0,018$	$0,493 \pm 0,01$	$0,495 \pm 0,011$	$0,507 \pm 0,011$	$0,013 \pm 0,001$
Naive Bayes	$0,041 \pm 0,006$	$0,085 \pm 0,005$	$0,064 \pm 0,003$	$0,163 \pm 0,013$	$0,055 \pm 0,002$
Linear SVM	$0,138 \pm 0,009$	$0,351 \pm 0,007$	$0,337 \pm 0,009$	$0,386 \pm 0,009$	$0,018 \pm 0,001$
Neural Net	$0,214 \pm 0,012$	$0,423 \pm 0,005$	$0,445 \pm 0,003$	$0,42 \pm 0,007$	$0,014 \pm 0,001$
RBF SVM	$0,08 \pm 0,005$	$0,066 \pm 0,003$	$0,169 \pm 0,006$	$0,048 \pm 0,003$	$0,013 \pm 0,0$
Adaboost	$0,395 \pm 0,018$	$0,516 \pm 0,009$	$0,658 \pm 0,013$	$0,446 \pm 0,009$	$0,01 \pm 0,0$
Random Forest	$0,237 \pm 0,018$	$0,268 \pm 0,019$	$0,617 \pm 0,032$	$0,195 \pm 0,013$	$0,011 \pm 0,0$

CUADRO 6.22. Resultados de Bibtex con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,043 \pm 0,005$	$0,062 \pm 0,006$	$0,136 \pm 0,025$	$0,065 \pm 0,005$	$0,021 \pm 0,0$
Decision Tree	$0,156 \pm 0,012$	$0,209 \pm 0,004$	$0,217 \pm 0,007$	$0,211 \pm 0,003$	$0,021 \pm 0,001$
Naive Bayes	$0,126 \pm 0,007$	$0,073 \pm 0,004$	$0,221 \pm 0,025$	$0,062 \pm 0,002$	$0,019 \pm 0,0$
Linear SVM	$0,23 \pm 0,014$	$0,272 \pm 0,007$	$0,376 \pm 0,013$	$0,249 \pm 0,005$	$0,016 \pm 0,0$
Neural Net	$0,138 \pm 0,012$	$0,167 \pm 0,007$	$0,18 \pm 0,003$	$0,168 \pm 0,009$	$0,023 \pm 0,001$
RBF SVM	$0,151 \pm 0,012$	$0,066 \pm 0,004$	$0,137 \pm 0,017$	$0,07 \pm 0,003$	$0,019 \pm 0,0$
Adaboost	$0,068 \pm 0,005$	$0,013 \pm 0,002$	$0,014 \pm 0,002$	$0,023 \pm 0,002$	$0,019 \pm 0,0$
Random Forest	$0,18 \pm 0,009$	$0,178 \pm 0,01$	$0,251 \pm 0,01$	$0,165 \pm 0,011$	$0,018 \pm 0,0$

CUADRO 6.23. Resultados de Bibtex con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,064 \pm 0,009$	$0,094 \pm 0,006$	$0,206 \pm 0,021$	$0,099 \pm 0,003$	$0,024 \pm 0,001$
Decision Tree	$0,565 \pm 0,01$	$0,388 \pm 0,012$	$0,404 \pm 0,012$	$0,391 \pm 0,011$	$0,017 \pm 0,001$
Naive Bayes	$0,177 \pm 0,011$	$0,111 \pm 0,009$	$0,322 \pm 0,025$	$0,091 \pm 0,007$	$0,018 \pm 0,0$
Linear SVM	$0,304 \pm 0,007$	$0,3 \pm 0,011$	$0,402 \pm 0,019$	$0,289 \pm 0,007$	$0,019 \pm 0,0$
Neural Net	$0,245 \pm 0,008$	$0,222 \pm 0,006$	$0,248 \pm 0,01$	$0,217 \pm 0,005$	$0,02 \pm 0,001$
RBF SVM	$0,175 \pm 0,006$	$0,099 \pm 0,004$	$0,28 \pm 0,028$	$0,086 \pm 0,003$	$0,018 \pm 0,0$
Adaboost	$0,104 \pm 0,007$	$0,021 \pm 0,004$	$0,038 \pm 0,011$	$0,025 \pm 0,003$	$0,02 \pm 0,0$
Random Forest	$0,572 \pm 0,014$	$0,431 \pm 0,004$	$0,511 \pm 0,006$	$0,407 \pm 0,005$	$0,015 \pm 0,0$

CUADRO 6.24. Resultados de Bibtex con Label Powerset- bin-rfl

6.5.6. Resultados de Genbase. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,918 \pm 0,014$	$0,616 \pm 0,032$	$0,636 \pm 0,029$	$0,605 \pm 0,035$	$0,005 \pm 0,0$
Decision Tree	$0,973 \pm 0,01$	$0,724 \pm 0,038$	$0,72 \pm 0,041$	$0,73 \pm 0,035$	$0,001 \pm 0,001$
Naive Bayes	$0,9 \pm 0,018$	$0,685 \pm 0,036$	$0,691 \pm 0,029$	$0,696 \pm 0,043$	$0,005 \pm 0,001$
Linear SVM	$0,973 \pm 0,014$	$0,724 \pm 0,039$	$0,72 \pm 0,041$	$0,731 \pm 0,038$	$0,001 \pm 0,001$
Neural Net	$0,968 \pm 0,006$	$0,726 \pm 0,039$	$0,718 \pm 0,037$	$0,739 \pm 0,041$	$0,001 \pm 0,001$
RBF SVM	$0,706 \pm 0,06$	$0,481 \pm 0,013$	$0,569 \pm 0,019$	$0,451 \pm 0,02$	$0,012 \pm 0,002$
Adaboost	$0,974 \pm 0,012$	$0,731 \pm 0,044$	$0,728 \pm 0,047$	$0,737 \pm 0,04$	$0,001 \pm 0,001$
Random Forest	$0,964 \pm 0,009$	$0,689 \pm 0,042$	$0,706 \pm 0,049$	$0,684 \pm 0,037$	$0,002 \pm 0,001$

CUADRO 6.25. Resultados de Genbase con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,927 \pm 0,012$	$0,63 \pm 0,042$	$0,663 \pm 0,046$	$0,612 \pm 0,042$	$0,005 \pm 0,001$
Decision Tree	$0,985 \pm 0,012$	$0,751 \pm 0,032$	$0,75 \pm 0,034$	$0,753 \pm 0,029$	$0,001 \pm 0,001$
Naive Bayes	$0,819 \pm 0,019$	$0,659 \pm 0,049$	$0,64 \pm 0,047$	$0,72 \pm 0,055$	$0,011 \pm 0,002$
Linear SVM	$0,927 \pm 0,038$	$0,719 \pm 0,037$	$0,71 \pm 0,035$	$0,742 \pm 0,042$	$0,004 \pm 0,001$
Neural Net	$0,965 \pm 0,015$	$0,739 \pm 0,037$	$0,738 \pm 0,031$	$0,746 \pm 0,046$	$0,002 \pm 0,001$
RBF SVM	$0,37 \pm 0,043$	$0,373 \pm 0,053$	$0,593 \pm 0,052$	$0,317 \pm 0,049$	$0,027 \pm 0,002$
Adaboost	$0,983 \pm 0,015$	$0,753 \pm 0,03$	$0,754 \pm 0,031$	$0,753 \pm 0,029$	$0,001 \pm 0,001$
Random Forest	$0,976 \pm 0,015$	$0,737 \pm 0,033$	$0,744 \pm 0,03$	$0,733 \pm 0,033$	$0,001 \pm 0,001$

CUADRO 6.26. Resultados de Genbase con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,932 \pm 0,022$	$0,65 \pm 0,022$	$0,673 \pm 0,017$	$0,641 \pm 0,026$	$0,004 \pm 0,002$
Decision Tree	$0,971 \pm 0,015$	$0,744 \pm 0,022$	$0,741 \pm 0,024$	$0,75 \pm 0,02$	$0,001 \pm 0,001$
Naive Bayes	$0,929 \pm 0,026$	$0,721 \pm 0,043$	$0,724 \pm 0,042$	$0,731 \pm 0,038$	$0,005 \pm 0,001$
Linear SVM	$0,971 \pm 0,014$	$0,751 \pm 0,034$	$0,748 \pm 0,035$	$0,76 \pm 0,032$	$0,002 \pm 0,001$
Neural Net	$0,964 \pm 0,009$	$0,753 \pm 0,03$	$0,745 \pm 0,033$	$0,769 \pm 0,027$	$0,002 \pm 0,0$
RBF SVM	$0,875 \pm 0,025$	$0,48 \pm 0,013$	$0,509 \pm 0,005$	$0,466 \pm 0,016$	$0,008 \pm 0,003$
Adaboost	$0,713 \pm 0,074$	$0,322 \pm 0,063$	$0,353 \pm 0,087$	$0,332 \pm 0,056$	$0,026 \pm 0,006$
Random Forest	$0,97 \pm 0,016$	$0,725 \pm 0,027$	$0,728 \pm 0,029$	$0,727 \pm 0,026$	$0,001 \pm 0,001$

CUADRO 6.27. Resultados de Genbase con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,935 \pm 0,02$	$0,603 \pm 0,036$	$0,636 \pm 0,028$	$0,587 \pm 0,039$	$0,005 \pm 0,001$
Decision Tree	$0,982 \pm 0,01$	$0,709 \pm 0,048$	$0,708 \pm 0,046$	$0,716 \pm 0,051$	$0,002 \pm 0,002$
Naive Bayes	$0,965 \pm 0,012$	$0,721 \pm 0,054$	$0,728 \pm 0,056$	$0,718 \pm 0,053$	$0,003 \pm 0,001$
Linear SVM	$0,976 \pm 0,003$	$0,723 \pm 0,05$	$0,728 \pm 0,047$	$0,724 \pm 0,055$	$0,002 \pm 0,001$
Neural Net	$0,964 \pm 0,011$	$0,691 \pm 0,059$	$0,696 \pm 0,075$	$0,701 \pm 0,05$	$0,004 \pm 0,002$
RBF SVM	$0,873 \pm 0,045$	$0,452 \pm 0,025$	$0,501 \pm 0,013$	$0,432 \pm 0,031$	$0,011 \pm 0,004$
Adaboost	$0,691 \pm 0,135$	$0,254 \pm 0,136$	$0,262 \pm 0,136$	$0,272 \pm 0,138$	$0,031 \pm 0,012$
Random Forest	$0,979 \pm 0,009$	$0,729 \pm 0,066$	$0,731 \pm 0,064$	$0,728 \pm 0,068$	$0,002 \pm 0,001$

CUADRO 6.28. Resultados de Genbase con Label Powerset- bin-rfl

6.5.7. Resultados de IMDB. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,051 \pm 0,003$	$0,033 \pm 0,002$	$0,198 \pm 0,048$	$0,031 \pm 0,001$	$0,081 \pm 0,001$
Decision Tree	$0,025 \pm 0,002$	$0,097 \pm 0,002$	$0,097 \pm 0,002$	$0,097 \pm 0,001$	$0,113 \pm 0,001$
Naive Bayes	$0,0 \pm 0,0$	$0,137 \pm 0,001$	$0,089 \pm 0,001$	$0,561 \pm 0,009$	$0,438 \pm 0,006$
Linear SVM	$0,02 \pm 0,002$	$0,104 \pm 0,004$	$0,102 \pm 0,003$	$0,11 \pm 0,006$	$0,117 \pm 0,001$
Neural Net	$0,034 \pm 0,002$	$0,096 \pm 0,002$	$0,122 \pm 0,002$	$0,082 \pm 0,001$	$0,096 \pm 0,001$
RBF SVM	$0,001 \pm 0,0$	$0,005 \pm 0,002$	$0,139 \pm 0,036$	$0,003 \pm 0,001$	$0,07 \pm 0,001$
Adaboost	$0,01 \pm 0,001$	$0,027 \pm 0,003$	$0,252 \pm 0,02$	$0,015 \pm 0,002$	$0,071 \pm 0,001$
Random Forest	$0,022 \pm 0,002$	$0,026 \pm 0,003$	$0,282 \pm 0,06$	$0,016 \pm 0,001$	$0,073 \pm 0,0$

CUADRO 6.29. Resultados de IMDB con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,077 \pm 0,005$	$0,052 \pm 0,002$	$0,124 \pm 0,015$	$0,052 \pm 0,002$	$0,092 \pm 0,001$
Decision Tree	$0,06 \pm 0,001$	$0,092 \pm 0,003$	$0,103 \pm 0,005$	$0,087 \pm 0,002$	$0,104 \pm 0,0$
Naive Bayes	$0,01 \pm 0,001$	$0,084 \pm 0,003$	$0,093 \pm 0,005$	$0,088 \pm 0,003$	$0,116 \pm 0,0$
Linear SVM	$0,039 \pm 0,004$	$0,094 \pm 0,003$	$0,089 \pm 0,004$	$0,105 \pm 0,003$	$0,125 \pm 0,002$
Neural Net	$0,044 \pm 0,002$	$0,097 \pm 0,002$	$0,103 \pm 0,003$	$0,093 \pm 0,002$	$0,108 \pm 0,001$
RBF SVM	$0,116 \pm 0,004$	$0,044 \pm 0,001$	$0,194 \pm 0,044$	$0,047 \pm 0,001$	$0,082 \pm 0,0$
Adaboost	$0,104 \pm 0,005$	$0,026 \pm 0,001$	$0,045 \pm 0,009$	$0,038 \pm 0,0$	$0,083 \pm 0,001$
Random Forest	$0,086 \pm 0,001$	$0,072 \pm 0,003$	$0,128 \pm 0,008$	$0,065 \pm 0,002$	$0,092 \pm 0,0$

CUADRO 6.30. Resultados de IMDB con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,019 \pm 0,002$	$0,074 \pm 0,003$	$0,097 \pm 0,004$	$0,083 \pm 0,004$	$0,116 \pm 0,0$
Decision Tree	$0,611 \pm 0,004$	$0,401 \pm 0,008$	$0,4 \pm 0,007$	$0,405 \pm 0,01$	$0,062 \pm 0,001$
Naive Bayes	$0,073 \pm 0,001$	$0,109 \pm 0,001$	$0,12 \pm 0,001$	$0,109 \pm 0,001$	$0,109 \pm 0,001$
Linear SVM	$0,086 \pm 0,005$	$0,124 \pm 0,004$	$0,117 \pm 0,004$	$0,137 \pm 0,003$	$0,121 \pm 0,001$
Neural Net	$0,102 \pm 0,003$	$0,141 \pm 0,004$	$0,149 \pm 0,004$	$0,135 \pm 0,004$	$0,103 \pm 0,0$
RBF SVM	$0,161 \pm 0,004$	$0,13 \pm 0,004$	$0,31 \pm 0,025$	$0,137 \pm 0,005$	$0,097 \pm 0,003$
Adaboost	$0,104 \pm 0,003$	$0,02 \pm 0,002$	$0,04 \pm 0,015$	$0,036 \pm 0,001$	$0,081 \pm 0,0$
Random Forest	$0,563 \pm 0,005$	$0,357 \pm 0,004$	$0,381 \pm 0,004$	$0,361 \pm 0,007$	$0,064 \pm 0,0$

CUADRO 6.31. Resultados de IMDB con Label Powerset- bin-rfl

6.5.8. Resultados de Medical. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,321 \pm 0,022$	$0,146 \pm 0,022$	$0,218 \pm 0,023$	$0,129 \pm 0,024$	$0,022 \pm 0,001$
Decision Tree	$0,635 \pm 0,041$	$0,401 \pm 0,047$	$0,409 \pm 0,05$	$0,412 \pm 0,045$	$0,012 \pm 0,001$
Naive Bayes	$0,157 \pm 0,015$	$0,176 \pm 0,016$	$0,194 \pm 0,027$	$0,189 \pm 0,023$	$0,036 \pm 0,002$
Linear SVM	$0,577 \pm 0,036$	$0,379 \pm 0,043$	$0,385 \pm 0,038$	$0,392 \pm 0,052$	$0,014 \pm 0,001$
Neural Net	$0,573 \pm 0,043$	$0,381 \pm 0,036$	$0,387 \pm 0,035$	$0,394 \pm 0,041$	$0,013 \pm 0,002$
RBF SVM	$0,147 \pm 0,022$	$0,071 \pm 0,008$	$0,161 \pm 0,011$	$0,051 \pm 0,008$	$0,023 \pm 0,001$
Adaboost	$0,644 \pm 0,033$	$0,404 \pm 0,025$	$0,419 \pm 0,032$	$0,407 \pm 0,024$	$0,011 \pm 0,001$
Random Forest	$0,492 \pm 0,023$	$0,219 \pm 0,027$	$0,295 \pm 0,032$	$0,19 \pm 0,026$	$0,014 \pm 0,0$

CUADRO 6.32. Resultados de Medical con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,251 \pm 0,024$	$0,168 \pm 0,008$	$0,265 \pm 0,027$	$0,14 \pm 0,007$	$0,02 \pm 0,001$
Decision Tree	$0,838 \pm 0,021$	$0,485 \pm 0,014$	$0,493 \pm 0,019$	$0,487 \pm 0,014$	$0,005 \pm 0,0$
Naive Bayes	$0,167 \pm 0,021$	$0,169 \pm 0,029$	$0,159 \pm 0,03$	$0,208 \pm 0,032$	$0,04 \pm 0,003$
Linear SVM	$0,413 \pm 0,039$	$0,345 \pm 0,027$	$0,325 \pm 0,026$	$0,397 \pm 0,036$	$0,021 \pm 0,002$
Neural Net	$0,584 \pm 0,066$	$0,408 \pm 0,011$	$0,407 \pm 0,008$	$0,431 \pm 0,015$	$0,013 \pm 0,002$
RBF SVM	$0,068 \pm 0,013$	$0,066 \pm 0,014$	$0,169 \pm 0,023$	$0,045 \pm 0,013$	$0,025 \pm 0,001$
Adaboost	$0,873 \pm 0,023$	$0,514 \pm 0,019$	$0,531 \pm 0,028$	$0,51 \pm 0,017$	$0,003 \pm 0,001$
Random Forest	$0,709 \pm 0,05$	$0,316 \pm 0,04$	$0,385 \pm 0,036$	$0,289 \pm 0,04$	$0,008 \pm 0,001$

CUADRO 6.33. Resultados de Medical con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,393 \pm 0,059$	$0,192 \pm 0,025$	$0,232 \pm 0,028$	$0,19 \pm 0,033$	$0,025 \pm 0,003$
Decision Tree	$0,651 \pm 0,022$	$0,314 \pm 0,011$	$0,325 \pm 0,008$	$0,326 \pm 0,013$	$0,014 \pm 0,001$
Naive Bayes	$0,308 \pm 0,029$	$0,189 \pm 0,023$	$0,215 \pm 0,027$	$0,19 \pm 0,022$	$0,03 \pm 0,001$
Linear SVM	$0,628 \pm 0,04$	$0,359 \pm 0,026$	$0,377 \pm 0,022$	$0,365 \pm 0,026$	$0,014 \pm 0,001$
Neural Net	$0,537 \pm 0,041$	$0,328 \pm 0,028$	$0,336 \pm 0,028$	$0,354 \pm 0,03$	$0,02 \pm 0,002$
RBF SVM	$0,31 \pm 0,03$	$0,079 \pm 0,022$	$0,126 \pm 0,035$	$0,081 \pm 0,018$	$0,03 \pm 0,002$
Adaboost	$0,235 \pm 0,058$	$0,038 \pm 0,008$	$0,042 \pm 0,01$	$0,054 \pm 0,008$	$0,034 \pm 0,004$
Random Forest	$0,607 \pm 0,035$	$0,28 \pm 0,014$	$0,299 \pm 0,023$	$0,281 \pm 0,014$	$0,015 \pm 0,001$

CUADRO 6.34. Resultados de Medical con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,469 \pm 0,043$	$0,242 \pm 0,018$	$0,285 \pm 0,017$	$0,253 \pm 0,02$	$0,024 \pm 0,002$
Decision Tree	$0,88 \pm 0,03$	$0,465 \pm 0,034$	$0,473 \pm 0,033$	$0,473 \pm 0,033$	$0,007 \pm 0,003$
Naive Bayes	$0,408 \pm 0,03$	$0,224 \pm 0,023$	$0,258 \pm 0,032$	$0,224 \pm 0,02$	$0,027 \pm 0,001$
Linear SVM	$0,681 \pm 0,028$	$0,367 \pm 0,021$	$0,378 \pm 0,029$	$0,381 \pm 0,028$	$0,014 \pm 0,002$
Neural Net	$0,663 \pm 0,04$	$0,398 \pm 0,018$	$0,406 \pm 0,018$	$0,415 \pm 0,02$	$0,014 \pm 0,002$
RBF SVM	$0,343 \pm 0,015$	$0,093 \pm 0,018$	$0,167 \pm 0,023$	$0,085 \pm 0,014$	$0,029 \pm 0,001$
Adaboost	$0,454 \pm 0,046$	$0,127 \pm 0,019$	$0,147 \pm 0,018$	$0,133 \pm 0,017$	$0,026 \pm 0,002$
Random Forest	$0,872 \pm 0,039$	$0,441 \pm 0,017$	$0,456 \pm 0,014$	$0,448 \pm 0,019$	$0,007 \pm 0,002$

CUADRO 6.35. Resultados de Medical con Label Powerset- bin-rfl

6.5.9. Resultados de Scene. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,139 \pm 0,067$	$0,173 \pm 0,043$	$0,357 \pm 0,063$	$0,152 \pm 0,074$	$0,206 \pm 0,025$
Decision Tree	$0,078 \pm 0,01$	$0,185 \pm 0,013$	$0,307 \pm 0,024$	$0,136 \pm 0,012$	$0,204 \pm 0,003$
Naive Bayes	$0,035 \pm 0,005$	$0,327 \pm 0,004$	$0,198 \pm 0,002$	$0,943 \pm 0,015$	$0,692 \pm 0,003$
Linear SVM	$0,066 \pm 0,012$	$0,167 \pm 0,02$	$0,324 \pm 0,055$	$0,117 \pm 0,013$	$0,197 \pm 0,006$
Neural Net	$0,083 \pm 0,011$	$0,147 \pm 0,02$	$0,335 \pm 0,075$	$0,105 \pm 0,016$	$0,183 \pm 0,006$
RBF SVM	$0,026 \pm 0,007$	$0,047 \pm 0,009$	$0,293 \pm 0,074$	$0,027 \pm 0,007$	$0,178 \pm 0,002$
Adaboost	$0,069 \pm 0,007$	$0,13 \pm 0,015$	$0,37 \pm 0,068$	$0,085 \pm 0,012$	$0,181 \pm 0,004$
Random Forest	$0,103 \pm 0,012$	$0,185 \pm 0,022$	$0,367 \pm 0,05$	$0,13 \pm 0,017$	$0,187 \pm 0,003$

CUADRO 6.36. Resultados de Scene con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,849 \pm 0,018$	$0,884 \pm 0,016$	$0,916 \pm 0,012$	$0,858 \pm 0,02$	$0,039 \pm 0,005$
Decision Tree	$0,989 \pm 0,007$	$0,992 \pm 0,004$	$0,992 \pm 0,004$	$0,993 \pm 0,006$	$0,003 \pm 0,001$
Naive Bayes	$0,08 \pm 0,049$	$0,42 \pm 0,018$	$0,464 \pm 0,018$	$0,552 \pm 0,049$	$0,307 \pm 0,026$
Linear SVM	$0,12 \pm 0,079$	$0,16 \pm 0,072$	$0,4 \pm 0,085$	$0,216 \pm 0,108$	$0,242 \pm 0,07$
Neural Net	$0,669 \pm 0,025$	$0,78 \pm 0,016$	$0,821 \pm 0,02$	$0,753 \pm 0,014$	$0,076 \pm 0,005$
RBF SVM	$0,372 \pm 0,023$	$0,435 \pm 0,016$	$0,653 \pm 0,046$	$0,396 \pm 0,009$	$0,113 \pm 0,006$
Adaboost	$0,995 \pm 0,003$	$0,997 \pm 0,002$	$0,997 \pm 0,002$	$0,997 \pm 0,003$	$0,001 \pm 0,001$
Random Forest	$0,998 \pm 0,002$	$0,999 \pm 0,001$	$0,999 \pm 0,001$	$0,999 \pm 0,002$	$0,0 \pm 0,0$

CUADRO 6.37. Resultados de Scene con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,642 \pm 0,012$	$0,722 \pm 0,006$	$0,81 \pm 0,014$	$0,675 \pm 0,005$	$0,094 \pm 0,003$
Decision Tree	$0,398 \pm 0,021$	$0,612 \pm 0,014$	$0,606 \pm 0,013$	$0,624 \pm 0,019$	$0,145 \pm 0,005$
Naive Bayes	$0,178 \pm 0,01$	$0,585 \pm 0,01$	$0,456 \pm 0,007$	$0,857 \pm 0,018$	$0,237 \pm 0,006$
Linear SVM	$0,451 \pm 0,023$	$0,64 \pm 0,022$	$0,635 \pm 0,021$	$0,65 \pm 0,023$	$0,131 \pm 0,008$
Neural Net	$0,566 \pm 0,021$	$0,727 \pm 0,015$	$0,739 \pm 0,012$	$0,719 \pm 0,019$	$0,1 \pm 0,005$
RBF SVM	$0,631 \pm 0,019$	$0,755 \pm 0,011$	$0,865 \pm 0,007$	$0,677 \pm 0,012$	$0,079 \pm 0,003$
Adaboost	$0,575 \pm 0,006$	$0,733 \pm 0,007$	$0,776 \pm 0,021$	$0,699 \pm 0,008$	$0,094 \pm 0,003$
Random Forest	$0,503 \pm 0,011$	$0,662 \pm 0,013$	$0,854 \pm 0,016$	$0,553 \pm 0,014$	$0,098 \pm 0,002$

CUADRO 6.38. Resultados de Scene con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,646 \pm 0,014$	$0,715 \pm 0,011$	$0,81 \pm 0,008$	$0,674 \pm 0,015$	$0,096 \pm 0,002$
Decision Tree	$0,413 \pm 0,024$	$0,626 \pm 0,016$	$0,624 \pm 0,02$	$0,63 \pm 0,013$	$0,136 \pm 0,007$
Naive Bayes	$0,162 \pm 0,018$	$0,579 \pm 0,009$	$0,473 \pm 0,013$	$0,793 \pm 0,014$	$0,228 \pm 0,007$
Linear SVM	$0,393 \pm 0,048$	$0,623 \pm 0,022$	$0,709 \pm 0,067$	$0,638 \pm 0,077$	$0,14 \pm 0,021$
Neural Net	$0,57 \pm 0,012$	$0,723 \pm 0,012$	$0,739 \pm 0,008$	$0,711 \pm 0,018$	$0,1 \pm 0,004$
RBF SVM	$0,638 \pm 0,017$	$0,764 \pm 0,011$	$0,894 \pm 0,006$	$0,672 \pm 0,013$	$0,075 \pm 0,004$
Adaboost	$0,598 \pm 0,007$	$0,753 \pm 0,005$	$0,806 \pm 0,013$	$0,71 \pm 0,009$	$0,085 \pm 0,001$
Random Forest	$0,517 \pm 0,019$	$0,685 \pm 0,013$	$0,875 \pm 0,015$	$0,574 \pm 0,007$	$0,092 \pm 0,005$

CUADRO 6.39. Resultados de Scene con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,261 \pm 0,023$	$0,253 \pm 0,021$	$0,361 \pm 0,033$	$0,282 \pm 0,02$	$0,248 \pm 0,008$
Decision Tree	$0,249 \pm 0,03$	$0,245 \pm 0,025$	$0,326 \pm 0,026$	$0,283 \pm 0,019$	$0,254 \pm 0,009$
Naive Bayes	$0,084 \pm 0,019$	$0,257 \pm 0,045$	$0,288 \pm 0,037$	$0,371 \pm 0,09$	$0,353 \pm 0,009$
Linear SVM	$0,246 \pm 0,025$	$0,246 \pm 0,015$	$0,335 \pm 0,025$	$0,282 \pm 0,014$	$0,255 \pm 0,006$
Neural Net	$0,279 \pm 0,035$	$0,251 \pm 0,033$	$0,322 \pm 0,074$	$0,303 \pm 0,021$	$0,243 \pm 0,012$
RBF SVM	$0,257 \pm 0,029$	$0,223 \pm 0,027$	$0,307 \pm 0,052$	$0,278 \pm 0,021$	$0,252 \pm 0,01$
Adaboost	$0,248 \pm 0,017$	$0,228 \pm 0,013$	$0,342 \pm 0,035$	$0,276 \pm 0,013$	$0,255 \pm 0,006$
Random Forest	$0,269 \pm 0,031$	$0,265 \pm 0,023$	$0,348 \pm 0,025$	$0,298 \pm 0,018$	$0,247 \pm 0,01$

CUADRO 6.40. Resultados de Scene con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,877 \pm 0,007$	$0,885 \pm 0,006$	$0,886 \pm 0,007$	$0,887 \pm 0,005$	$0,042 \pm 0,003$
Decision Tree	$0,999 \pm 0,002$	$0,999 \pm 0,003$	$0,999 \pm 0,002$	$0,998 \pm 0,003$	$0,0 \pm 0,001$
Naive Bayes	$0,94 \pm 0,009$	$0,939 \pm 0,008$	$0,939 \pm 0,008$	$0,941 \pm 0,006$	$0,022 \pm 0,002$
Linear SVM	$0,948 \pm 0,012$	$0,951 \pm 0,011$	$0,951 \pm 0,011$	$0,952 \pm 0,011$	$0,018 \pm 0,004$
Neural Net	$0,748 \pm 0,047$	$0,767 \pm 0,045$	$0,792 \pm 0,04$	$0,761 \pm 0,038$	$0,083 \pm 0,014$
RBF SVM	$0,811 \pm 0,012$	$0,824 \pm 0,007$	$0,837 \pm 0,007$	$0,837 \pm 0,006$	$0,066 \pm 0,004$
Adaboost	$0,597 \pm 0,172$	$0,53 \pm 0,212$	$0,598 \pm 0,203$	$0,582 \pm 0,173$	$0,138 \pm 0,061$
Random Forest	$0,999 \pm 0,002$	$0,999 \pm 0,003$	$0,999 \pm 0,002$	$0,998 \pm 0,003$	$0,0 \pm 0,001$

CUADRO 6.41. Resultados de Scene con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,671 \pm 0,025$	$0,716 \pm 0,018$	$0,779 \pm 0,021$	$0,696 \pm 0,015$	$0,102 \pm 0,007$
Decision Tree	$0,516 \pm 0,022$	$0,583 \pm 0,02$	$0,585 \pm 0,022$	$0,588 \pm 0,021$	$0,154 \pm 0,007$
Naive Bayes	$0,546 \pm 0,027$	$0,658 \pm 0,02$	$0,667 \pm 0,019$	$0,681 \pm 0,017$	$0,132 \pm 0,009$
Linear SVM	$0,659 \pm 0,017$	$0,722 \pm 0,017$	$0,729 \pm 0,02$	$0,719 \pm 0,018$	$0,103 \pm 0,006$
Neural Net	$0,665 \pm 0,014$	$0,729 \pm 0,011$	$0,733 \pm 0,015$	$0,727 \pm 0,011$	$0,1 \pm 0,004$
RBF SVM	$0,743 \pm 0,019$	$0,787 \pm 0,016$	$0,812 \pm 0,016$	$0,77 \pm 0,017$	$0,076 \pm 0,005$
Adaboost	$0,302 \pm 0,038$	$0,236 \pm 0,071$	$0,272 \pm 0,056$	$0,335 \pm 0,059$	$0,229 \pm 0,011$
Random Forest	$0,666 \pm 0,017$	$0,715 \pm 0,016$	$0,743 \pm 0,01$	$0,696 \pm 0,021$	$0,103 \pm 0,006$

CUADRO 6.42. Resultados de Scene con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,696 \pm 0,031$	$0,723 \pm 0,023$	$0,754 \pm 0,02$	$0,725 \pm 0,024$	$0,103 \pm 0,011$
Decision Tree	$0,583 \pm 0,025$	$0,608 \pm 0,02$	$0,61 \pm 0,023$	$0,61 \pm 0,02$	$0,143 \pm 0,009$
Naive Bayes	$0,647 \pm 0,028$	$0,67 \pm 0,027$	$0,706 \pm 0,034$	$0,666 \pm 0,021$	$0,119 \pm 0,01$
Linear SVM	$0,7 \pm 0,03$	$0,728 \pm 0,023$	$0,734 \pm 0,027$	$0,73 \pm 0,02$	$0,102 \pm 0,011$
Neural Net	$0,702 \pm 0,021$	$0,747 \pm 0,014$	$0,751 \pm 0,017$	$0,747 \pm 0,013$	$0,093 \pm 0,007$
RBF SVM	$0,791 \pm 0,027$	$0,809 \pm 0,022$	$0,81 \pm 0,024$	$0,811 \pm 0,019$	$0,071 \pm 0,009$
Adaboost	$0,288 \pm 0,034$	$0,248 \pm 0,036$	$0,293 \pm 0,076$	$0,344 \pm 0,021$	$0,241 \pm 0,011$
Random Forest	$0,705 \pm 0,016$	$0,733 \pm 0,014$	$0,743 \pm 0,013$	$0,73 \pm 0,016$	$0,099 \pm 0,006$

CUADRO 6.43. Resultados de Scene con Label Powerset- tf-rfl

6.5.10. Resultados de Yeast. [H]

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,0 \pm 0,0$	$0,126 \pm 0,103$	$0,098 \pm 0,08$	$0,214 \pm 0,175$	$0,321 \pm 0,012$
Decision Tree	$0,014 \pm 0,004$	$0,122 \pm 0,001$	$0,107 \pm 0,002$	$0,143 \pm 0,0$	$0,232 \pm 0,004$
Naive Bayes	$0,0 \pm 0,0$	$0,18 \pm 0,029$	$0,144 \pm 0,079$	$0,557 \pm 0,029$	$0,629 \pm 0,075$
Linear SVM	$0,014 \pm 0,004$	$0,122 \pm 0,001$	$0,107 \pm 0,002$	$0,143 \pm 0,0$	$0,232 \pm 0,004$
Neural Net	$0,014 \pm 0,004$	$0,122 \pm 0,001$	$0,121 \pm 0,028$	$0,143 \pm 0,0$	$0,232 \pm 0,004$
RBF SVM	$0,014 \pm 0,004$	$0,122 \pm 0,001$	$0,107 \pm 0,002$	$0,143 \pm 0,0$	$0,232 \pm 0,004$
Adaboost	$0,014 \pm 0,004$	$0,122 \pm 0,001$	$0,107 \pm 0,002$	$0,143 \pm 0,0$	$0,232 \pm 0,004$
Random Forest	$0,014 \pm 0,004$	$0,122 \pm 0,001$	$0,107 \pm 0,002$	$0,143 \pm 0,0$	$0,232 \pm 0,004$

CUADRO 6.44. Resultados de Yeast con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,913 \pm 0,007$	$0,893 \pm 0,011$	$0,918 \pm 0,01$	$0,877 \pm 0,022$	$0,032 \pm 0,003$
Decision Tree	$0,964 \pm 0,008$	$0,948 \pm 0,015$	$0,954 \pm 0,015$	$0,943 \pm 0,017$	$0,015 \pm 0,004$
Naive Bayes	$0,0 \pm 0,0$	$0,381 \pm 0,004$	$0,459 \pm 0,011$	$0,39 \pm 0,011$	$0,274 \pm 0,003$
Linear SVM	$0,0 \pm 0,0$	$0,251 \pm 0,07$	$0,271 \pm 0,036$	$0,342 \pm 0,121$	$0,267 \pm 0,038$
Neural Net	$0,018 \pm 0,013$	$0,334 \pm 0,019$	$0,551 \pm 0,05$	$0,294 \pm 0,014$	$0,18 \pm 0,006$
RBF SVM	$0,027 \pm 0,009$	$0,285 \pm 0,004$	$0,54 \pm 0,024$	$0,239 \pm 0,005$	$0,181 \pm 0,002$
Adaboost	$0,712 \pm 0,013$	$0,781 \pm 0,009$	$0,907 \pm 0,023$	$0,717 \pm 0,011$	$0,063 \pm 0,003$
Random Forest	$0,948 \pm 0,005$	$0,941 \pm 0,014$	$0,953 \pm 0,014$	$0,931 \pm 0,018$	$0,017 \pm 0,003$

CUADRO 6.45. Resultados de Yeast con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,205 \pm 0,013$	$0,431 \pm 0,017$	$0,537 \pm 0,048$	$0,404 \pm 0,017$	$0,205 \pm 0,004$
Decision Tree	$0,05 \pm 0,009$	$0,384 \pm 0,009$	$0,384 \pm 0,007$	$0,387 \pm 0,012$	$0,28 \pm 0,005$
Naive Bayes	$0,093 \pm 0,014$	$0,45 \pm 0,008$	$0,42 \pm 0,004$	$0,529 \pm 0,014$	$0,302 \pm 0,009$
Linear SVM	$0,152 \pm 0,009$	$0,338 \pm 0,008$	$0,364 \pm 0,015$	$0,338 \pm 0,013$	$0,201 \pm 0,006$
Neural Net	$0,083 \pm 0,004$	$0,42 \pm 0,009$	$0,427 \pm 0,006$	$0,419 \pm 0,016$	$0,251 \pm 0,002$
RBF SVM	$0,196 \pm 0,015$	$0,375 \pm 0,014$	$0,648 \pm 0,068$	$0,348 \pm 0,01$	$0,187 \pm 0,006$
Adaboost	$0,108 \pm 0,008$	$0,408 \pm 0,009$	$0,469 \pm 0,004$	$0,383 \pm 0,013$	$0,221 \pm 0,003$
Random Forest	$0,107 \pm 0,008$	$0,344 \pm 0,007$	$0,561 \pm 0,019$	$0,298 \pm 0,008$	$0,211 \pm 0,003$

CUADRO 6.46. Resultados de Yeast con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,131 \pm 0,01$	$0,374 \pm 0,018$	$0,521 \pm 0,022$	$0,297 \pm 0,015$	$0,214 \pm 0,009$
Decision Tree	$0,077 \pm 0,007$	$0,448 \pm 0,007$	$0,44 \pm 0,008$	$0,46 \pm 0,008$	$0,23 \pm 0,008$
Naive Bayes	$0,0 \pm 0,0$	$0,396 \pm 0,012$	$0,443 \pm 0,009$	$0,414 \pm 0,024$	$0,27 \pm 0,007$
Linear SVM	$0,022 \pm 0,004$	$0,387 \pm 0,007$	$0,384 \pm 0,012$	$0,404 \pm 0,014$	$0,297 \pm 0,009$
Neural Net	$0,134 \pm 0,013$	$0,461 \pm 0,015$	$0,479 \pm 0,023$	$0,45 \pm 0,015$	$0,2 \pm 0,008$
RBF SVM	$0,195 \pm 0,017$	$0,382 \pm 0,007$	$0,627 \pm 0,017$	$0,331 \pm 0,007$	$0,153 \pm 0,008$
Adaboost	$0,163 \pm 0,019$	$0,46 \pm 0,013$	$0,545 \pm 0,015$	$0,416 \pm 0,012$	$0,167 \pm 0,006$
Random Forest	$0,151 \pm 0,019$	$0,415 \pm 0,014$	$0,585 \pm 0,024$	$0,358 \pm 0,01$	$0,166 \pm 0,009$

CUADRO 6.47. Resultados de Yeast con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,034 \pm 0,018$	$0,184 \pm 0,017$	$0,145 \pm 0,011$	$0,314 \pm 0,057$	$0,326 \pm 0,04$
Decision Tree	$0,089 \pm 0,006$	$0,197 \pm 0,005$	$0,158 \pm 0,005$	$0,286 \pm 0,0$	$0,273 \pm 0,008$
Naive Bayes	$0,076 \pm 0,029$	$0,175 \pm 0,048$	$0,181 \pm 0,044$	$0,257 \pm 0,057$	$0,283 \pm 0,025$
Linear SVM	$0,089 \pm 0,006$	$0,197 \pm 0,005$	$0,158 \pm 0,005$	$0,286 \pm 0,0$	$0,273 \pm 0,008$
Neural Net	$0,089 \pm 0,006$	$0,197 \pm 0,005$	$0,158 \pm 0,005$	$0,286 \pm 0,0$	$0,273 \pm 0,008$
RBF SVM	$0,089 \pm 0,006$	$0,197 \pm 0,005$	$0,158 \pm 0,005$	$0,286 \pm 0,0$	$0,273 \pm 0,008$
Adaboost	$0,089 \pm 0,006$	$0,197 \pm 0,005$	$0,158 \pm 0,005$	$0,286 \pm 0,0$	$0,273 \pm 0,008$
Random Forest	$0,089 \pm 0,006$	$0,195 \pm 0,004$	$0,156 \pm 0,004$	$0,286 \pm 0,0$	$0,276 \pm 0,008$

CUADRO 6.48. Resultados de Yeast con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,788 \pm 0,032$	$0,746 \pm 0,042$	$0,782 \pm 0,025$	$0,745 \pm 0,05$	$0,098 \pm 0,016$
Decision Tree	$0,77 \pm 0,026$	$0,735 \pm 0,038$	$0,783 \pm 0,022$	$0,722 \pm 0,045$	$0,102 \pm 0,012$
Naive Bayes	$0,77 \pm 0,026$	$0,735 \pm 0,038$	$0,783 \pm 0,022$	$0,722 \pm 0,045$	$0,102 \pm 0,012$
Linear SVM	$0,755 \pm 0,021$	$0,725 \pm 0,03$	$0,763 \pm 0,038$	$0,717 \pm 0,037$	$0,107 \pm 0,007$
Neural Net	$0,456 \pm 0,033$	$0,499 \pm 0,028$	$0,542 \pm 0,037$	$0,527 \pm 0,061$	$0,187 \pm 0,015$
RBF SVM	$0,394 \pm 0,011$	$0,467 \pm 0,016$	$0,501 \pm 0,022$	$0,47 \pm 0,035$	$0,204 \pm 0,005$
Adaboost	$0,32 \pm 0,089$	$0,429 \pm 0,057$	$0,482 \pm 0,055$	$0,473 \pm 0,057$	$0,237 \pm 0,03$
Random Forest	$0,779 \pm 0,028$	$0,731 \pm 0,03$	$0,765 \pm 0,019$	$0,728 \pm 0,037$	$0,102 \pm 0,013$

CUADRO 6.49. Resultados de Yeast con Label Powerset-
bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,219 \pm 0,013$	$0,426 \pm 0,009$	$0,466 \pm 0,011$	$0,411 \pm 0,01$	$0,24 \pm 0,006$
Decision Tree	$0,139 \pm 0,006$	$0,377 \pm 0,014$	$0,38 \pm 0,015$	$0,376 \pm 0,015$	$0,28 \pm 0,006$
Naive Bayes	$0,194 \pm 0,013$	$0,443 \pm 0,011$	$0,453 \pm 0,01$	$0,437 \pm 0,018$	$0,24 \pm 0,006$
Linear SVM	$0,213 \pm 0,017$	$0,453 \pm 0,009$	$0,469 \pm 0,009$	$0,44 \pm 0,011$	$0,234 \pm 0,007$
Neural Net	$0,176 \pm 0,009$	$0,413 \pm 0,007$	$0,424 \pm 0,006$	$0,406 \pm 0,008$	$0,253 \pm 0,007$
RBF SVM	$0,278 \pm 0,014$	$0,399 \pm 0,012$	$0,601 \pm 0,049$	$0,397 \pm 0,01$	$0,199 \pm 0,008$
Adaboost	$0,139 \pm 0,006$	$0,225 \pm 0,002$	$0,255 \pm 0,056$	$0,298 \pm 0,001$	$0,259 \pm 0,005$
Random Forest	$0,192 \pm 0,012$	$0,391 \pm 0,022$	$0,43 \pm 0,026$	$0,368 \pm 0,019$	$0,246 \pm 0,005$

CUADRO 6.50. Resultados de Yeast con Label Powerset-
tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,13 \pm 0,015$	$0,393 \pm 0,013$	$0,433 \pm 0,021$	$0,444 \pm 0,011$	$0,311 \pm 0,01$
Decision Tree	$0,201 \pm 0,013$	$0,435 \pm 0,016$	$0,432 \pm 0,011$	$0,44 \pm 0,022$	$0,265 \pm 0,004$
Naive Bayes	$0,377 \pm 0,022$	$0,527 \pm 0,023$	$0,544 \pm 0,033$	$0,517 \pm 0,018$	$0,204 \pm 0,012$
Linear SVM	$0,157 \pm 0,012$	$0,43 \pm 0,011$	$0,41 \pm 0,014$	$0,464 \pm 0,011$	$0,283 \pm 0,008$
Neural Net	$0,197 \pm 0,025$	$0,45 \pm 0,022$	$0,455 \pm 0,031$	$0,449 \pm 0,019$	$0,25 \pm 0,011$
RBF SVM	$0,437 \pm 0,034$	$0,521 \pm 0,015$	$0,57 \pm 0,036$	$0,494 \pm 0,005$	$0,191 \pm 0,012$
Adaboost	$0,137 \pm 0,014$	$0,278 \pm 0,029$	$0,306 \pm 0,049$	$0,309 \pm 0,007$	$0,267 \pm 0,017$
Random Forest	$0,262 \pm 0,021$	$0,449 \pm 0,013$	$0,471 \pm 0,013$	$0,436 \pm 0,016$	$0,241 \pm 0,008$

CUADRO 6.51. Resultados de Yeast con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,498 \pm 0,05$	$0,081 \pm 0,008$	$0,272 \pm 0,024$	$0,066 \pm 0,007$	$0,03 \pm 0,003$
Decision Tree	$0,449 \pm 0,003$	$0,245 \pm 0,013$	$0,248 \pm 0,017$	$0,249 \pm 0,012$	$0,035 \pm 0,001$
Naive Bayes	$0,277 \pm 0,013$	$0,082 \pm 0,003$	$0,075 \pm 0,005$	$0,1 \pm 0,006$	$0,063 \pm 0,002$
Linear SVM	$0,483 \pm 0,006$	$0,263 \pm 0,019$	$0,256 \pm 0,019$	$0,282 \pm 0,022$	$0,032 \pm 0,001$
Neural Net	$0,56 \pm 0,011$	$0,368 \pm 0,02$	$0,484 \pm 0,025$	$0,314 \pm 0,018$	$0,024 \pm 0,001$
RBF SVM	$0,562 \pm 0,012$	$0,08 \pm 0,005$	$0,285 \pm 0,008$	$0,062 \pm 0,003$	$0,027 \pm 0,001$
Adaboost	$0,562 \pm 0,012$	$0,274 \pm 0,024$	$0,429 \pm 0,046$	$0,212 \pm 0,019$	$0,025 \pm 0,001$
Random Forest	$0,565 \pm 0,006$	$0,154 \pm 0,004$	$0,436 \pm 0,032$	$0,11 \pm 0,003$	$0,025 \pm 0,001$

CUADRO 6.52. Resultados de Business con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,546 \pm 0,009$	$0,099 \pm 0,007$	$0,419 \pm 0,046$	$0,071 \pm 0,004$	$0,027 \pm 0,001$
Decision Tree	$0,776 \pm 0,005$	$0,435 \pm 0,028$	$0,445 \pm 0,039$	$0,442 \pm 0,027$	$0,015 \pm 0,001$
Naive Bayes	$0,293 \pm 0,004$	$0,093 \pm 0,01$	$0,089 \pm 0,015$	$0,11 \pm 0,008$	$0,061 \pm 0,001$
Linear SVM	$0,396 \pm 0,007$	$0,264 \pm 0,008$	$0,207 \pm 0,007$	$0,418 \pm 0,019$	$0,041 \pm 0,001$
Neural Net	$0,615 \pm 0,004$	$0,412 \pm 0,008$	$0,402 \pm 0,013$	$0,455 \pm 0,025$	$0,022 \pm 0,0$
RBF SVM	$0,591 \pm 0,011$	$0,063 \pm 0,004$	$0,242 \pm 0,061$	$0,056 \pm 0,002$	$0,025 \pm 0,001$
Adaboost	$0,787 \pm 0,006$	$0,481 \pm 0,016$	$0,649 \pm 0,029$	$0,403 \pm 0,01$	$0,012 \pm 0,0$
Random Forest	$0,797 \pm 0,001$	$0,297 \pm 0,022$	$0,598 \pm 0,048$	$0,233 \pm 0,014$	$0,012 \pm 0,0$

CUADRO 6.53. Resultados de Business con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,543 \pm 0,011$	$0,089 \pm 0,011$	$0,309 \pm 0,029$	$0,067 \pm 0,007$	$0,028 \pm 0,001$
Decision Tree	$0,467 \pm 0,007$	$0,289 \pm 0,012$	$0,301 \pm 0,01$	$0,287 \pm 0,013$	$0,033 \pm 0,001$
Naive Bayes	$0,287 \pm 0,008$	$0,089 \pm 0,006$	$0,083 \pm 0,008$	$0,108 \pm 0,006$	$0,063 \pm 0,001$
Linear SVM	$0,418 \pm 0,007$	$0,242 \pm 0,012$	$0,212 \pm 0,01$	$0,305 \pm 0,022$	$0,038 \pm 0,001$
Neural Net	$0,519 \pm 0,006$	$0,331 \pm 0,008$	$0,369 \pm 0,023$	$0,315 \pm 0,003$	$0,028 \pm 0,0$
RBF SVM	$0,546 \pm 0,012$	$0,062 \pm 0,008$	$0,253 \pm 0,034$	$0,051 \pm 0,005$	$0,028 \pm 0,001$
Adaboost	$0,559 \pm 0,011$	$0,296 \pm 0,009$	$0,443 \pm 0,024$	$0,238 \pm 0,009$	$0,025 \pm 0,001$
Random Forest	$0,568 \pm 0,014$	$0,17 \pm 0,007$	$0,439 \pm 0,034$	$0,121 \pm 0,006$	$0,025 \pm 0,001$

CUADRO 6.54. Resultados de Business con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,555 \pm 0,007$	$0,106 \pm 0,007$	$0,457 \pm 0,052$	$0,076 \pm 0,004$	$0,026 \pm 0,001$
Decision Tree	$0,7 \pm 0,01$	$0,366 \pm 0,009$	$0,366 \pm 0,008$	$0,375 \pm 0,014$	$0,019 \pm 0,001$
Naive Bayes	$0,297 \pm 0,018$	$0,094 \pm 0,008$	$0,089 \pm 0,01$	$0,112 \pm 0,007$	$0,061 \pm 0,0$
Linear SVM	$0,376 \pm 0,011$	$0,251 \pm 0,015$	$0,198 \pm 0,013$	$0,391 \pm 0,023$	$0,043 \pm 0,001$
Neural Net	$0,552 \pm 0,008$	$0,361 \pm 0,013$	$0,34 \pm 0,024$	$0,422 \pm 0,015$	$0,027 \pm 0,001$
RBF SVM	$0,569 \pm 0,007$	$0,057 \pm 0,004$	$0,214 \pm 0,039$	$0,052 \pm 0,003$	$0,026 \pm 0,001$
Adaboost	$0,738 \pm 0,011$	$0,389 \pm 0,024$	$0,54 \pm 0,044$	$0,319 \pm 0,018$	$0,015 \pm 0,001$
Random Forest	$0,763 \pm 0,007$	$0,265 \pm 0,02$	$0,536 \pm 0,045$	$0,209 \pm 0,015$	$0,015 \pm 0,0$

CUADRO 6.55. Resultados de Business con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,499 \pm 0,039$	$0,093 \pm 0,013$	$0,269 \pm 0,065$	$0,077 \pm 0,006$	$0,035 \pm 0,005$
Decision Tree	$0,503 \pm 0,007$	$0,18 \pm 0,008$	$0,198 \pm 0,014$	$0,169 \pm 0,006$	$0,033 \pm 0,0$
Naive Bayes	$0,418 \pm 0,006$	$0,096 \pm 0,005$	$0,147 \pm 0,014$	$0,084 \pm 0,005$	$0,037 \pm 0,001$
Linear SVM	$0,502 \pm 0,016$	$0,187 \pm 0,017$	$0,163 \pm 0,017$	$0,263 \pm 0,023$	$0,051 \pm 0,004$
Neural Net	$0,555 \pm 0,009$	$0,249 \pm 0,032$	$0,306 \pm 0,061$	$0,229 \pm 0,029$	$0,03 \pm 0,001$
RBF SVM	$0,555 \pm 0,006$	$0,072 \pm 0,004$	$0,291 \pm 0,027$	$0,056 \pm 0,002$	$0,027 \pm 0,001$
Adaboost	$0,538 \pm 0,007$	$0,049 \pm 0,004$	$0,061 \pm 0,007$	$0,048 \pm 0,002$	$0,029 \pm 0,001$
Random Forest	$0,596 \pm 0,008$	$0,191 \pm 0,016$	$0,438 \pm 0,025$	$0,143 \pm 0,011$	$0,025 \pm 0,001$

CUADRO 6.56. Resultados de Business con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,2 \pm 0,017$	$0,137 \pm 0,017$	$0,146 \pm 0,024$	$0,225 \pm 0,025$	$0,084 \pm 0,01$
Decision Tree	$0,905 \pm 0,005$	$0,487 \pm 0,015$	$0,51 \pm 0,037$	$0,492 \pm 0,014$	$0,012 \pm 0,0$
Naive Bayes	$0,441 \pm 0,005$	$0,124 \pm 0,007$	$0,209 \pm 0,007$	$0,105 \pm 0,006$	$0,036 \pm 0,001$
Linear SVM	$0,647 \pm 0,007$	$0,259 \pm 0,018$	$0,237 \pm 0,019$	$0,323 \pm 0,021$	$0,036 \pm 0,002$
Neural Net	$0,675 \pm 0,017$	$0,337 \pm 0,027$	$0,384 \pm 0,032$	$0,326 \pm 0,023$	$0,024 \pm 0,002$
RBF SVM	$0,585 \pm 0,006$	$0,092 \pm 0,01$	$0,331 \pm 0,027$	$0,069 \pm 0,006$	$0,026 \pm 0,001$
Adaboost	$0,535 \pm 0,005$	$0,033 \pm 0,004$	$0,036 \pm 0,013$	$0,035 \pm 0,003$	$0,029 \pm 0,0$
Random Forest	$0,888 \pm 0,005$	$0,496 \pm 0,017$	$0,557 \pm 0,02$	$0,479 \pm 0,028$	$0,013 \pm 0,001$

CUADRO 6.57. Resultados de Business con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,486 \pm 0,044$	$0,096 \pm 0,014$	$0,241 \pm 0,051$	$0,085 \pm 0,006$	$0,038 \pm 0,007$
Decision Tree	$0,502 \pm 0,004$	$0,19 \pm 0,013$	$0,213 \pm 0,019$	$0,18 \pm 0,012$	$0,033 \pm 0,0$
Naive Bayes	$0,415 \pm 0,009$	$0,105 \pm 0,015$	$0,162 \pm 0,025$	$0,091 \pm 0,011$	$0,037 \pm 0,001$
Linear SVM	$0,47 \pm 0,018$	$0,195 \pm 0,022$	$0,168 \pm 0,021$	$0,276 \pm 0,042$	$0,051 \pm 0,003$
Neural Net	$0,54 \pm 0,015$	$0,237 \pm 0,023$	$0,283 \pm 0,035$	$0,219 \pm 0,014$	$0,031 \pm 0,002$
RBF SVM	$0,542 \pm 0,011$	$0,042 \pm 0,003$	$0,207 \pm 0,028$	$0,039 \pm 0,002$	$0,028 \pm 0,001$
Adaboost	$0,538 \pm 0,008$	$0,05 \pm 0,003$	$0,064 \pm 0,007$	$0,048 \pm 0,002$	$0,029 \pm 0,0$
Random Forest	$0,596 \pm 0,009$	$0,198 \pm 0,013$	$0,454 \pm 0,045$	$0,146 \pm 0,01$	$0,024 \pm 0,001$

CUADRO 6.58. Resultados de Business con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,327 \pm 0,018$	$0,156 \pm 0,014$	$0,173 \pm 0,019$	$0,215 \pm 0,022$	$0,069 \pm 0,006$
Decision Tree	$0,834 \pm 0,009$	$0,371 \pm 0,018$	$0,392 \pm 0,027$	$0,364 \pm 0,018$	$0,018 \pm 0,001$
Naive Bayes	$0,441 \pm 0,004$	$0,121 \pm 0,01$	$0,199 \pm 0,039$	$0,103 \pm 0,007$	$0,036 \pm 0,001$
Linear SVM	$0,599 \pm 0,004$	$0,248 \pm 0,015$	$0,222 \pm 0,017$	$0,323 \pm 0,018$	$0,038 \pm 0,001$
Neural Net	$0,634 \pm 0,017$	$0,301 \pm 0,023$	$0,329 \pm 0,037$	$0,311 \pm 0,024$	$0,028 \pm 0,003$
RBF SVM	$0,555 \pm 0,008$	$0,063 \pm 0,005$	$0,287 \pm 0,047$	$0,051 \pm 0,003$	$0,028 \pm 0,0$
Adaboost	$0,534 \pm 0,009$	$0,031 \pm 0,0$	$0,029 \pm 0,0$	$0,033 \pm 0,0$	$0,029 \pm 0,0$
Random Forest	$0,715 \pm 0,007$	$0,282 \pm 0,012$	$0,383 \pm 0,018$	$0,243 \pm 0,014$	$0,022 \pm 0,001$

CUADRO 6.59. Resultados de Business con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,351 \pm 0,01$	$0,098 \pm 0,005$	$0,428 \pm 0,078$	$0,074 \pm 0,002$	$0,041 \pm 0,0$
Decision Tree	$0,285 \pm 0,012$	$0,246 \pm 0,005$	$0,241 \pm 0,004$	$0,26 \pm 0,013$	$0,049 \pm 0,001$
Naive Bayes	$0,144 \pm 0,007$	$0,089 \pm 0,01$	$0,077 \pm 0,01$	$0,118 \pm 0,013$	$0,083 \pm 0,002$
Linear SVM	$0,357 \pm 0,006$	$0,289 \pm 0,011$	$0,286 \pm 0,009$	$0,307 \pm 0,019$	$0,043 \pm 0,001$
Neural Net	$0,421 \pm 0,007$	$0,372 \pm 0,013$	$0,485 \pm 0,018$	$0,316 \pm 0,015$	$0,034 \pm 0,0$
RBF SVM	$0,347 \pm 0,008$	$0,113 \pm 0,006$	$0,465 \pm 0,061$	$0,082 \pm 0,005$	$0,035 \pm 0,0$
Adaboost	$0,381 \pm 0,016$	$0,289 \pm 0,023$	$0,463 \pm 0,033$	$0,224 \pm 0,019$	$0,034 \pm 0,001$
Random Forest	$0,365 \pm 0,01$	$0,188 \pm 0,016$	$0,574 \pm 0,053$	$0,13 \pm 0,009$	$0,033 \pm 0,001$

CUADRO 6.60. Resultados de Computers con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,336 \pm 0,007$	$0,101 \pm 0,012$	$0,438 \pm 0,071$	$0,071 \pm 0,007$	$0,039 \pm 0,001$
Decision Tree	$0,657 \pm 0,013$	$0,488 \pm 0,019$	$0,491 \pm 0,021$	$0,493 \pm 0,021$	$0,021 \pm 0,001$
Naive Bayes	$0,152 \pm 0,008$	$0,088 \pm 0,004$	$0,072 \pm 0,004$	$0,121 \pm 0,005$	$0,084 \pm 0,001$
Linear SVM	$0,252 \pm 0,005$	$0,279 \pm 0,009$	$0,218 \pm 0,008$	$0,44 \pm 0,015$	$0,058 \pm 0,001$
Neural Net	$0,39 \pm 0,01$	$0,431 \pm 0,016$	$0,42 \pm 0,02$	$0,464 \pm 0,015$	$0,037 \pm 0,001$
RBF SVM	$0,346 \pm 0,009$	$0,063 \pm 0,007$	$0,321 \pm 0,001$	$0,05 \pm 0,005$	$0,039 \pm 0,001$
Adaboost	$0,596 \pm 0,012$	$0,486 \pm 0,022$	$0,652 \pm 0,024$	$0,403 \pm 0,026$	$0,021 \pm 0,001$
Random Forest	$0,629 \pm 0,006$	$0,403 \pm 0,011$	$0,688 \pm 0,013$	$0,311 \pm 0,013$	$0,019 \pm 0,0$

CUADRO 6.61. Resultados de Computers con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,328 \pm 0,022$	$0,107 \pm 0,01$	$0,473 \pm 0,03$	$0,076 \pm 0,007$	$0,04 \pm 0,001$
Decision Tree	$0,296 \pm 0,007$	$0,266 \pm 0,022$	$0,266 \pm 0,022$	$0,272 \pm 0,026$	$0,047 \pm 0,001$
Naive Bayes	$0,145 \pm 0,008$	$0,104 \pm 0,014$	$0,104 \pm 0,026$	$0,13 \pm 0,01$	$0,083 \pm 0,003$
Linear SVM	$0,299 \pm 0,011$	$0,259 \pm 0,014$	$0,229 \pm 0,013$	$0,32 \pm 0,025$	$0,051 \pm 0,001$
Neural Net	$0,377 \pm 0,007$	$0,351 \pm 0,035$	$0,389 \pm 0,043$	$0,333 \pm 0,035$	$0,039 \pm 0,001$
RBF SVM	$0,335 \pm 0,006$	$0,097 \pm 0,007$	$0,404 \pm 0,043$	$0,073 \pm 0,006$	$0,037 \pm 0,001$
Adaboost	$0,387 \pm 0,009$	$0,308 \pm 0,023$	$0,477 \pm 0,029$	$0,241 \pm 0,022$	$0,034 \pm 0,001$
Random Forest	$0,377 \pm 0,007$	$0,196 \pm 0,011$	$0,533 \pm 0,043$	$0,135 \pm 0,006$	$0,033 \pm 0,001$

CUADRO 6.62. Resultados de Computers con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,33 \pm 0,003$	$0,098 \pm 0,011$	$0,461 \pm 0,038$	$0,069 \pm 0,007$	$0,039 \pm 0,001$
Decision Tree	$0,53 \pm 0,016$	$0,424 \pm 0,006$	$0,422 \pm 0,01$	$0,434 \pm 0,019$	$0,028 \pm 0,001$
Naive Bayes	$0,158 \pm 0,005$	$0,089 \pm 0,007$	$0,072 \pm 0,005$	$0,126 \pm 0,015$	$0,084 \pm 0,001$
Linear SVM	$0,235 \pm 0,004$	$0,261 \pm 0,004$	$0,204 \pm 0,002$	$0,399 \pm 0,015$	$0,06 \pm 0,001$
Neural Net	$0,364 \pm 0,005$	$0,397 \pm 0,005$	$0,372 \pm 0,006$	$0,451 \pm 0,025$	$0,04 \pm 0,001$
RBF SVM	$0,334 \pm 0,002$	$0,055 \pm 0,009$	$0,345 \pm 0,036$	$0,046 \pm 0,005$	$0,041 \pm 0,001$
Adaboost	$0,513 \pm 0,017$	$0,441 \pm 0,02$	$0,6 \pm 0,028$	$0,362 \pm 0,018$	$0,025 \pm 0,001$
Random Forest	$0,517 \pm 0,004$	$0,345 \pm 0,012$	$0,654 \pm 0,011$	$0,257 \pm 0,012$	$0,024 \pm 0,0$

CUADRO 6.63. Resultados de Computers con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,366 \pm 0,005$	$0,12 \pm 0,006$	$0,39 \pm 0,02$	$0,09 \pm 0,005$	$0,042 \pm 0,0$
Decision Tree	$0,367 \pm 0,009$	$0,192 \pm 0,01$	$0,21 \pm 0,013$	$0,181 \pm 0,012$	$0,049 \pm 0,001$
Naive Bayes	$0,344 \pm 0,012$	$0,141 \pm 0,008$	$0,262 \pm 0,013$	$0,113 \pm 0,007$	$0,046 \pm 0,0$
Linear SVM	$0,398 \pm 0,005$	$0,21 \pm 0,013$	$0,183 \pm 0,009$	$0,286 \pm 0,025$	$0,069 \pm 0,002$
Neural Net	$0,401 \pm 0,012$	$0,218 \pm 0,018$	$0,244 \pm 0,028$	$0,212 \pm 0,016$	$0,049 \pm 0,001$
RBF SVM	$0,423 \pm 0,012$	$0,128 \pm 0,004$	$0,425 \pm 0,073$	$0,101 \pm 0,004$	$0,039 \pm 0,001$
Adaboost	$0,331 \pm 0,006$	$0,021 \pm 0,001$	$0,022 \pm 0,012$	$0,03 \pm 0,0$	$0,044 \pm 0,0$
Random Forest	$0,452 \pm 0,008$	$0,196 \pm 0,006$	$0,392 \pm 0,044$	$0,155 \pm 0,006$	$0,038 \pm 0,001$

CUADRO 6.64. Resultados de Computers con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,15 \pm 0,009$	$0,144 \pm 0,005$	$0,218 \pm 0,031$	$0,18 \pm 0,007$	$0,079 \pm 0,002$
Decision Tree	$0,855 \pm 0,005$	$0,531 \pm 0,015$	$0,54 \pm 0,021$	$0,528 \pm 0,015$	$0,018 \pm 0,001$
Naive Bayes	$0,389 \pm 0,009$	$0,177 \pm 0,015$	$0,292 \pm 0,021$	$0,142 \pm 0,011$	$0,044 \pm 0,001$
Linear SVM	$0,556 \pm 0,004$	$0,288 \pm 0,014$	$0,267 \pm 0,019$	$0,34 \pm 0,01$	$0,048 \pm 0,001$
Neural Net	$0,559 \pm 0,018$	$0,351 \pm 0,036$	$0,405 \pm 0,051$	$0,328 \pm 0,04$	$0,035 \pm 0,002$
RBF SVM	$0,458 \pm 0,005$	$0,14 \pm 0,002$	$0,419 \pm 0,076$	$0,11 \pm 0,003$	$0,037 \pm 0,0$
Adaboost	$0,331 \pm 0,012$	$0,021 \pm 0,0$	$0,016 \pm 0,0$	$0,03 \pm 0,0$	$0,044 \pm 0,001$
Random Forest	$0,831 \pm 0,009$	$0,567 \pm 0,024$	$0,605 \pm 0,033$	$0,556 \pm 0,027$	$0,018 \pm 0,001$

CUADRO 6.65. Resultados de Computers con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,376 \pm 0,006$	$0,138 \pm 0,015$	$0,336 \pm 0,015$	$0,106 \pm 0,013$	$0,042 \pm 0,001$
Decision Tree	$0,366 \pm 0,005$	$0,207 \pm 0,012$	$0,225 \pm 0,014$	$0,198 \pm 0,017$	$0,048 \pm 0,0$
Naive Bayes	$0,329 \pm 0,007$	$0,147 \pm 0,014$	$0,261 \pm 0,026$	$0,117 \pm 0,01$	$0,048 \pm 0,001$
Linear SVM	$0,36 \pm 0,012$	$0,214 \pm 0,019$	$0,193 \pm 0,02$	$0,276 \pm 0,022$	$0,066 \pm 0,002$
Neural Net	$0,389 \pm 0,01$	$0,215 \pm 0,02$	$0,243 \pm 0,021$	$0,203 \pm 0,022$	$0,048 \pm 0,001$
RBF SVM	$0,39 \pm 0,004$	$0,107 \pm 0,007$	$0,357 \pm 0,042$	$0,085 \pm 0,006$	$0,04 \pm 0,001$
Adaboost	$0,331 \pm 0,002$	$0,021 \pm 0,0$	$0,016 \pm 0,0$	$0,03 \pm 0,0$	$0,044 \pm 0,001$
Random Forest	$0,456 \pm 0,007$	$0,198 \pm 0,009$	$0,396 \pm 0,021$	$0,157 \pm 0,011$	$0,038 \pm 0,001$

CUADRO 6.66. Resultados de Computers con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,257 \pm 0,024$	$0,179 \pm 0,021$	$0,232 \pm 0,025$	$0,195 \pm 0,027$	$0,063 \pm 0,004$
Decision Tree	$0,74 \pm 0,012$	$0,429 \pm 0,024$	$0,438 \pm 0,034$	$0,431 \pm 0,019$	$0,027 \pm 0,001$
Naive Bayes	$0,37 \pm 0,005$	$0,179 \pm 0,014$	$0,287 \pm 0,035$	$0,145 \pm 0,011$	$0,045 \pm 0,001$
Linear SVM	$0,492 \pm 0,004$	$0,263 \pm 0,011$	$0,242 \pm 0,013$	$0,317 \pm 0,013$	$0,053 \pm 0,001$
Neural Net	$0,517 \pm 0,015$	$0,306 \pm 0,017$	$0,339 \pm 0,029$	$0,299 \pm 0,017$	$0,041 \pm 0,002$
RBF SVM	$0,414 \pm 0,002$	$0,119 \pm 0,01$	$0,343 \pm 0,015$	$0,094 \pm 0,009$	$0,039 \pm 0,0$
Adaboost	$0,331 \pm 0,006$	$0,021 \pm 0,0$	$0,016 \pm 0,0$	$0,03 \pm 0,0$	$0,044 \pm 0,001$
Random Forest	$0,621 \pm 0,004$	$0,324 \pm 0,015$	$0,441 \pm 0,017$	$0,283 \pm 0,016$	$0,031 \pm 0,001$

CUADRO 6.67. Resultados de Computers con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,082 \pm 0,01$	$0,059 \pm 0,007$	$0,201 \pm 0,013$	$0,046 \pm 0,007$	$0,054 \pm 0,003$
Decision Tree	$0,185 \pm 0,005$	$0,223 \pm 0,013$	$0,224 \pm 0,018$	$0,232 \pm 0,009$	$0,053 \pm 0,001$
Naive Bayes	$0,089 \pm 0,007$	$0,083 \pm 0,001$	$0,072 \pm 0,006$	$0,114 \pm 0,002$	$0,087 \pm 0,001$
Linear SVM	$0,242 \pm 0,008$	$0,236 \pm 0,012$	$0,233 \pm 0,01$	$0,247 \pm 0,014$	$0,048 \pm 0,0$
Neural Net	$0,297 \pm 0,004$	$0,304 \pm 0,014$	$0,409 \pm 0,037$	$0,259 \pm 0,009$	$0,04 \pm 0,001$
RBF SVM	$0,065 \pm 0,004$	$0,056 \pm 0,006$	$0,313 \pm 0,015$	$0,033 \pm 0,004$	$0,041 \pm 0,0$
Adaboost	$0,233 \pm 0,008$	$0,251 \pm 0,013$	$0,424 \pm 0,026$	$0,187 \pm 0,01$	$0,037 \pm 0,0$
Random Forest	$0,187 \pm 0,011$	$0,132 \pm 0,012$	$0,326 \pm 0,033$	$0,09 \pm 0,011$	$0,038 \pm 0,0$

CUADRO 6.68. Resultados de Education con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,09 \pm 0,004$	$0,088 \pm 0,006$	$0,351 \pm 0,014$	$0,052 \pm 0,004$	$0,042 \pm 0,0$
Decision Tree	$0,608 \pm 0,013$	$0,387 \pm 0,019$	$0,392 \pm 0,029$	$0,397 \pm 0,017$	$0,023 \pm 0,001$
Naive Bayes	$0,099 \pm 0,005$	$0,092 \pm 0,005$	$0,088 \pm 0,013$	$0,121 \pm 0,006$	$0,086 \pm 0,002$
Linear SVM	$0,171 \pm 0,008$	$0,21 \pm 0,012$	$0,168 \pm 0,008$	$0,324 \pm 0,017$	$0,063 \pm 0,001$
Neural Net	$0,261 \pm 0,007$	$0,284 \pm 0,009$	$0,266 \pm 0,009$	$0,328 \pm 0,009$	$0,047 \pm 0,001$
RBF SVM	$0,022 \pm 0,003$	$0,019 \pm 0,003$	$0,229 \pm 0,024$	$0,01 \pm 0,002$	$0,043 \pm 0,001$
Adaboost	$0,547 \pm 0,007$	$0,442 \pm 0,027$	$0,634 \pm 0,06$	$0,36 \pm 0,023$	$0,024 \pm 0,001$
Random Forest	$0,571 \pm 0,008$	$0,255 \pm 0,005$	$0,413 \pm 0,013$	$0,202 \pm 0,006$	$0,023 \pm 0,001$

CUADRO 6.69. Resultados de Education con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,079 \pm 0,011$	$0,054 \pm 0,004$	$0,207 \pm 0,019$	$0,034 \pm 0,002$	$0,046 \pm 0,001$
Decision Tree	$0,2 \pm 0,008$	$0,248 \pm 0,018$	$0,253 \pm 0,022$	$0,262 \pm 0,021$	$0,05 \pm 0,001$
Naive Bayes	$0,086 \pm 0,007$	$0,09 \pm 0,003$	$0,086 \pm 0,014$	$0,118 \pm 0,004$	$0,087 \pm 0,001$
Linear SVM	$0,202 \pm 0,009$	$0,207 \pm 0,012$	$0,188 \pm 0,009$	$0,248 \pm 0,023$	$0,055 \pm 0,0$
Neural Net	$0,267 \pm 0,005$	$0,281 \pm 0,017$	$0,301 \pm 0,023$	$0,278 \pm 0,02$	$0,045 \pm 0,0$
RBF SVM	$0,027 \pm 0,001$	$0,031 \pm 0,004$	$0,306 \pm 0,022$	$0,018 \pm 0,003$	$0,043 \pm 0,0$
Adaboost	$0,241 \pm 0,01$	$0,27 \pm 0,018$	$0,44 \pm 0,033$	$0,207 \pm 0,017$	$0,036 \pm 0,001$
Random Forest	$0,203 \pm 0,005$	$0,149 \pm 0,006$	$0,365 \pm 0,024$	$0,101 \pm 0,004$	$0,037 \pm 0,0$

CUADRO 6.70. Resultados de Education con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,091 \pm 0,009$	$0,081 \pm 0,009$	$0,319 \pm 0,038$	$0,049 \pm 0,005$	$0,042 \pm 0,001$
Decision Tree	$0,415 \pm 0,01$	$0,309 \pm 0,005$	$0,309 \pm 0,009$	$0,317 \pm 0,006$	$0,035 \pm 0,0$
Naive Bayes	$0,102 \pm 0,004$	$0,09 \pm 0,008$	$0,077 \pm 0,01$	$0,12 \pm 0,01$	$0,086 \pm 0,002$
Linear SVM	$0,154 \pm 0,006$	$0,209 \pm 0,004$	$0,168 \pm 0,003$	$0,338 \pm 0,005$	$0,064 \pm 0,001$
Neural Net	$0,232 \pm 0,009$	$0,261 \pm 0,009$	$0,231 \pm 0,009$	$0,331 \pm 0,017$	$0,051 \pm 0,001$
RBF SVM	$0,012 \pm 0,001$	$0,013 \pm 0,002$	$0,204 \pm 0,034$	$0,007 \pm 0,001$	$0,044 \pm 0,0$
Adaboost	$0,384 \pm 0,008$	$0,333 \pm 0,012$	$0,508 \pm 0,017$	$0,26 \pm 0,014$	$0,03 \pm 0,0$
Random Forest	$0,367 \pm 0,013$	$0,202 \pm 0,007$	$0,359 \pm 0,015$	$0,151 \pm 0,007$	$0,03 \pm 0,001$

CUADRO 6.71. Resultados de Education con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,116 \pm 0,005$	$0,068 \pm 0,004$	$0,152 \pm 0,02$	$0,07 \pm 0,004$	$0,069 \pm 0,002$
Decision Tree	$0,297 \pm 0,008$	$0,179 \pm 0,017$	$0,194 \pm 0,017$	$0,171 \pm 0,018$	$0,051 \pm 0,001$
Naive Bayes	$0,234 \pm 0,006$	$0,1 \pm 0,012$	$0,176 \pm 0,021$	$0,082 \pm 0,008$	$0,054 \pm 0,001$
Linear SVM	$0,281 \pm 0,008$	$0,17 \pm 0,012$	$0,156 \pm 0,015$	$0,215 \pm 0,013$	$0,068 \pm 0,002$
Neural Net	$0,31 \pm 0,016$	$0,198 \pm 0,025$	$0,231 \pm 0,031$	$0,194 \pm 0,024$	$0,053 \pm 0,003$
RBF SVM	$0,265 \pm 0,007$	$0,063 \pm 0,008$	$0,263 \pm 0,013$	$0,059 \pm 0,006$	$0,052 \pm 0,001$
Adaboost	$0,235 \pm 0,006$	$0,032 \pm 0,002$	$0,062 \pm 0,002$	$0,041 \pm 0,001$	$0,054 \pm 0,0$
Random Forest	$0,362 \pm 0,009$	$0,183 \pm 0,008$	$0,272 \pm 0,017$	$0,152 \pm 0,009$	$0,044 \pm 0,001$

CUADRO 6.72. Resultados de Education con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,121 \pm 0,006$	$0,127 \pm 0,006$	$0,169 \pm 0,008$	$0,166 \pm 0,005$	$0,083 \pm 0,001$
Decision Tree	$0,848 \pm 0,009$	$0,516 \pm 0,026$	$0,53 \pm 0,033$	$0,52 \pm 0,019$	$0,018 \pm 0,001$
Naive Bayes	$0,275 \pm 0,007$	$0,119 \pm 0,014$	$0,205 \pm 0,04$	$0,097 \pm 0,009$	$0,051 \pm 0,001$
Linear SVM	$0,451 \pm 0,005$	$0,226 \pm 0,011$	$0,214 \pm 0,009$	$0,262 \pm 0,016$	$0,051 \pm 0,001$
Neural Net	$0,488 \pm 0,025$	$0,285 \pm 0,025$	$0,325 \pm 0,039$	$0,282 \pm 0,025$	$0,04 \pm 0,004$
RBF SVM	$0,311 \pm 0,008$	$0,092 \pm 0,005$	$0,334 \pm 0,014$	$0,077 \pm 0,004$	$0,049 \pm 0,001$
Adaboost	$0,217 \pm 0,007$	$0,014 \pm 0,0$	$0,009 \pm 0,0$	$0,03 \pm 0,0$	$0,056 \pm 0,0$
Random Forest	$0,793 \pm 0,008$	$0,474 \pm 0,048$	$0,489 \pm 0,055$	$0,494 \pm 0,054$	$0,021 \pm 0,001$

CUADRO 6.73. Resultados de Education con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,155 \pm 0,003$	$0,094 \pm 0,014$	$0,198 \pm 0,032$	$0,08 \pm 0,012$	$0,062 \pm 0,002$
Decision Tree	$0,309 \pm 0,012$	$0,19 \pm 0,017$	$0,217 \pm 0,016$	$0,178 \pm 0,018$	$0,05 \pm 0,001$
Naive Bayes	$0,231 \pm 0,011$	$0,101 \pm 0,005$	$0,164 \pm 0,023$	$0,084 \pm 0,003$	$0,054 \pm 0,001$
Linear SVM	$0,251 \pm 0,007$	$0,168 \pm 0,009$	$0,151 \pm 0,009$	$0,217 \pm 0,017$	$0,069 \pm 0,001$
Neural Net	$0,284 \pm 0,014$	$0,185 \pm 0,021$	$0,203 \pm 0,027$	$0,186 \pm 0,021$	$0,056 \pm 0,002$
RBF SVM	$0,236 \pm 0,011$	$0,042 \pm 0,007$	$0,258 \pm 0,022$	$0,046 \pm 0,005$	$0,054 \pm 0,001$
Adaboost	$0,235 \pm 0,01$	$0,032 \pm 0,001$	$0,062 \pm 0,001$	$0,041 \pm 0,001$	$0,054 \pm 0,001$
Random Forest	$0,378 \pm 0,011$	$0,181 \pm 0,012$	$0,287 \pm 0,032$	$0,15 \pm 0,008$	$0,042 \pm 0,001$

CUADRO 6.74. Resultados de Education con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,16 \pm 0,007$	$0,128 \pm 0,008$	$0,182 \pm 0,012$	$0,159 \pm 0,013$	$0,075 \pm 0,002$
Decision Tree	$0,665 \pm 0,009$	$0,332 \pm 0,008$	$0,352 \pm 0,006$	$0,33 \pm 0,01$	$0,031 \pm 0,001$
Naive Bayes	$0,26 \pm 0,01$	$0,126 \pm 0,007$	$0,217 \pm 0,025$	$0,102 \pm 0,006$	$0,052 \pm 0,001$
Linear SVM	$0,373 \pm 0,01$	$0,197 \pm 0,008$	$0,182 \pm 0,01$	$0,236 \pm 0,01$	$0,057 \pm 0,001$
Neural Net	$0,417 \pm 0,012$	$0,266 \pm 0,022$	$0,313 \pm 0,024$	$0,256 \pm 0,024$	$0,046 \pm 0,001$
RBF SVM	$0,252 \pm 0,003$	$0,056 \pm 0,004$	$0,28 \pm 0,004$	$0,055 \pm 0,003$	$0,053 \pm 0,0$
Adaboost	$0,217 \pm 0,004$	$0,014 \pm 0,0$	$0,009 \pm 0,0$	$0,03 \pm 0,0$	$0,056 \pm 0,0$
Random Forest	$0,542 \pm 0,003$	$0,299 \pm 0,011$	$0,392 \pm 0,011$	$0,265 \pm 0,014$	$0,035 \pm 0,001$

CUADRO 6.75. Resultados de Education con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,14 \pm 0,009$	$0,124 \pm 0,009$	$0,423 \pm 0,041$	$0,087 \pm 0,007$	$0,064 \pm 0,001$
Decision Tree	$0,295 \pm 0,007$	$0,34 \pm 0,016$	$0,334 \pm 0,017$	$0,362 \pm 0,024$	$0,066 \pm 0,001$
Naive Bayes	$0,168 \pm 0,005$	$0,151 \pm 0,003$	$0,124 \pm 0,003$	$0,208 \pm 0,005$	$0,131 \pm 0,002$
Linear SVM	$0,394 \pm 0,012$	$0,372 \pm 0,011$	$0,374 \pm 0,013$	$0,392 \pm 0,018$	$0,056 \pm 0,001$
Neural Net	$0,475 \pm 0,015$	$0,461 \pm 0,015$	$0,545 \pm 0,031$	$0,418 \pm 0,019$	$0,044 \pm 0,001$
RBF SVM	$0,215 \pm 0,006$	$0,151 \pm 0,006$	$0,558 \pm 0,008$	$0,104 \pm 0,004$	$0,054 \pm 0,0$
Adaboost	$0,371 \pm 0,005$	$0,361 \pm 0,023$	$0,537 \pm 0,039$	$0,285 \pm 0,026$	$0,047 \pm 0,0$
Random Forest	$0,351 \pm 0,008$	$0,259 \pm 0,005$	$0,572 \pm 0,015$	$0,188 \pm 0,005$	$0,047 \pm 0,001$

CUADRO 6.76. Resultados de Entertainment con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,203 \pm 0,004$	$0,181 \pm 0,013$	$0,59 \pm 0,026$	$0,124 \pm 0,007$	$0,057 \pm 0,001$
Decision Tree	$0,711 \pm 0,005$	$0,567 \pm 0,023$	$0,568 \pm 0,021$	$0,577 \pm 0,03$	$0,025 \pm 0,001$
Naive Bayes	$0,201 \pm 0,005$	$0,165 \pm 0,012$	$0,152 \pm 0,013$	$0,207 \pm 0,011$	$0,118 \pm 0,001$
Linear SVM	$0,333 \pm 0,007$	$0,357 \pm 0,016$	$0,308 \pm 0,014$	$0,452 \pm 0,021$	$0,068 \pm 0,001$
Neural Net	$0,427 \pm 0,008$	$0,448 \pm 0,025$	$0,455 \pm 0,037$	$0,467 \pm 0,024$	$0,051 \pm 0,001$
RBF SVM	$0,117 \pm 0,004$	$0,079 \pm 0,003$	$0,531 \pm 0,047$	$0,046 \pm 0,002$	$0,06 \pm 0,001$
Adaboost	$0,636 \pm 0,014$	$0,587 \pm 0,016$	$0,779 \pm 0,012$	$0,489 \pm 0,016$	$0,027 \pm 0,001$
Random Forest	$0,685 \pm 0,008$	$0,462 \pm 0,005$	$0,656 \pm 0,031$	$0,389 \pm 0,006$	$0,026 \pm 0,001$

CUADRO 6.77. Resultados de Entertainment con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,16 \pm 0,011$	$0,15 \pm 0,02$	$0,533 \pm 0,044$	$0,102 \pm 0,012$	$0,061 \pm 0,001$
Decision Tree	$0,314 \pm 0,007$	$0,374 \pm 0,024$	$0,383 \pm 0,023$	$0,377 \pm 0,029$	$0,062 \pm 0,001$
Naive Bayes	$0,173 \pm 0,002$	$0,154 \pm 0,002$	$0,126 \pm 0,002$	$0,212 \pm 0,008$	$0,13 \pm 0,002$
Linear SVM	$0,343 \pm 0,005$	$0,349 \pm 0,014$	$0,324 \pm 0,013$	$0,4 \pm 0,025$	$0,066 \pm 0,001$
Neural Net	$0,422 \pm 0,009$	$0,429 \pm 0,005$	$0,461 \pm 0,022$	$0,425 \pm 0,014$	$0,051 \pm 0,001$
RBF SVM	$0,149 \pm 0,007$	$0,121 \pm 0,007$	$0,55 \pm 0,033$	$0,083 \pm 0,004$	$0,058 \pm 0,001$
Adaboost	$0,383 \pm 0,009$	$0,373 \pm 0,015$	$0,55 \pm 0,037$	$0,293 \pm 0,011$	$0,046 \pm 0,001$
Random Forest	$0,363 \pm 0,006$	$0,276 \pm 0,013$	$0,573 \pm 0,03$	$0,201 \pm 0,009$	$0,046 \pm 0,0$

CUADRO 6.78. Resultados de Entertainment con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	0,201 \pm 0,006	0,172 \pm 0,008	0,569 \pm 0,033	0,118 \pm 0,004	0,058 \pm 0,001
Decision Tree	0,541 \pm 0,011	0,497 \pm 0,014	0,498 \pm 0,026	0,516 \pm 0,025	0,039 \pm 0,001
Naive Bayes	0,201 \pm 0,006	0,167 \pm 0,01	0,149 \pm 0,006	0,215 \pm 0,019	0,118 \pm 0,002
Linear SVM	0,302 \pm 0,005	0,343 \pm 0,012	0,296 \pm 0,009	0,435 \pm 0,022	0,072 \pm 0,0
Neural Net	0,393 \pm 0,004	0,409 \pm 0,02	0,389 \pm 0,038	0,456 \pm 0,013	0,057 \pm 0,001
RBF SVM	0,076 \pm 0,005	0,059 \pm 0,008	0,542 \pm 0,057	0,033 \pm 0,005	0,063 \pm 0,001
Adaboost	0,519 \pm 0,004	0,496 \pm 0,019	0,665 \pm 0,024	0,41 \pm 0,022	0,034 \pm 0,0
Random Forest	0,544 \pm 0,003	0,4 \pm 0,005	0,602 \pm 0,021	0,319 \pm 0,004	0,033 \pm 0,0

CUADRO 6.79. Resultados de Entertainment con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	0,23 \pm 0,022	0,16 \pm 0,009	0,246 \pm 0,026	0,157 \pm 0,009	0,089 \pm 0,002
Decision Tree	0,427 \pm 0,009	0,298 \pm 0,018	0,317 \pm 0,023	0,285 \pm 0,016	0,066 \pm 0,001
Naive Bayes	0,36 \pm 0,012	0,178 \pm 0,01	0,27 \pm 0,023	0,153 \pm 0,008	0,072 \pm 0,002
Linear SVM	0,457 \pm 0,009	0,271 \pm 0,016	0,23 \pm 0,022	0,362 \pm 0,019	0,091 \pm 0,004
Neural Net	0,469 \pm 0,014	0,31 \pm 0,021	0,317 \pm 0,032	0,32 \pm 0,03	0,064 \pm 0,004
RBF SVM	0,399 \pm 0,01	0,166 \pm 0,007	0,518 \pm 0,014	0,147 \pm 0,004	0,067 \pm 0,001
Adaboost	0,277 \pm 0,006	0,064 \pm 0,001	0,086 \pm 0,021	0,084 \pm 0,001	0,081 \pm 0,001
Random Forest	0,485 \pm 0,01	0,283 \pm 0,012	0,407 \pm 0,025	0,243 \pm 0,01	0,057 \pm 0,001

CUADRO 6.80. Resultados de Entertainment con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	0,189 \pm 0,008	0,164 \pm 0,012	0,241 \pm 0,022	0,241 \pm 0,029	0,114 \pm 0,005
Decision Tree	0,881 \pm 0,007	0,624 \pm 0,017	0,655 \pm 0,017	0,62 \pm 0,018	0,021 \pm 0,001
Naive Bayes	0,407 \pm 0,013	0,211 \pm 0,007	0,318 \pm 0,02	0,179 \pm 0,005	0,068 \pm 0,002
Linear SVM	0,607 \pm 0,009	0,356 \pm 0,014	0,323 \pm 0,022	0,425 \pm 0,017	0,062 \pm 0,003
Neural Net	0,629 \pm 0,011	0,462 \pm 0,028	0,498 \pm 0,028	0,461 \pm 0,027	0,046 \pm 0,003
RBF SVM	0,46 \pm 0,01	0,208 \pm 0,007	0,581 \pm 0,003	0,172 \pm 0,005	0,062 \pm 0,001
Adaboost	0,249 \pm 0,008	0,049 \pm 0,01	0,092 \pm 0,043	0,071 \pm 0,007	0,086 \pm 0,001
Random Forest	0,85 \pm 0,007	0,628 \pm 0,029	0,653 \pm 0,038	0,637 \pm 0,03	0,024 \pm 0,001

CUADRO 6.81. Resultados de Entertainment con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,259 \pm 0,011$	$0,173 \pm 0,014$	$0,272 \pm 0,018$	$0,167 \pm 0,011$	$0,084 \pm 0,001$
Decision Tree	$0,428 \pm 0,012$	$0,299 \pm 0,012$	$0,316 \pm 0,014$	$0,289 \pm 0,013$	$0,065 \pm 0,002$
Naive Bayes	$0,361 \pm 0,008$	$0,189 \pm 0,019$	$0,273 \pm 0,019$	$0,165 \pm 0,02$	$0,073 \pm 0,001$
Linear SVM	$0,412 \pm 0,009$	$0,269 \pm 0,005$	$0,224 \pm 0,017$	$0,386 \pm 0,031$	$0,099 \pm 0,006$
Neural Net	$0,417 \pm 0,015$	$0,293 \pm 0,031$	$0,306 \pm 0,04$	$0,303 \pm 0,026$	$0,07 \pm 0,002$
RBF SVM	$0,349 \pm 0,007$	$0,14 \pm 0,009$	$0,516 \pm 0,012$	$0,128 \pm 0,005$	$0,073 \pm 0,002$
Adaboost	$0,277 \pm 0,006$	$0,064 \pm 0,001$	$0,062 \pm 0,0$	$0,084 \pm 0,002$	$0,081 \pm 0,002$
Random Forest	$0,498 \pm 0,01$	$0,291 \pm 0,014$	$0,406 \pm 0,016$	$0,252 \pm 0,013$	$0,056 \pm 0,002$

CUADRO 6.82. Resultados de Entertainment con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,237 \pm 0,011$	$0,19 \pm 0,015$	$0,274 \pm 0,03$	$0,238 \pm 0,017$	$0,101 \pm 0,002$
Decision Tree	$0,733 \pm 0,005$	$0,468 \pm 0,016$	$0,476 \pm 0,009$	$0,475 \pm 0,029$	$0,037 \pm 0,0$
Naive Bayes	$0,397 \pm 0,013$	$0,215 \pm 0,023$	$0,309 \pm 0,038$	$0,183 \pm 0,018$	$0,068 \pm 0,002$
Linear SVM	$0,532 \pm 0,009$	$0,324 \pm 0,027$	$0,28 \pm 0,03$	$0,418 \pm 0,024$	$0,077 \pm 0,006$
Neural Net	$0,563 \pm 0,013$	$0,414 \pm 0,026$	$0,409 \pm 0,047$	$0,452 \pm 0,027$	$0,059 \pm 0,008$
RBF SVM	$0,376 \pm 0,01$	$0,159 \pm 0,003$	$0,541 \pm 0,019$	$0,139 \pm 0,003$	$0,071 \pm 0,001$
Adaboost	$0,277 \pm 0,01$	$0,066 \pm 0,01$	$0,091 \pm 0,038$	$0,083 \pm 0,006$	$0,082 \pm 0,001$
Random Forest	$0,681 \pm 0,012$	$0,457 \pm 0,021$	$0,549 \pm 0,029$	$0,418 \pm 0,023$	$0,039 \pm 0,001$

CUADRO 6.83. Resultados de Entertainment con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,318 \pm 0,007$	$0,051 \pm 0,005$	$0,249 \pm 0,022$	$0,046 \pm 0,003$	$0,048 \pm 0,001$
Decision Tree	$0,318 \pm 0,007$	$0,299 \pm 0,017$	$0,304 \pm 0,017$	$0,302 \pm 0,02$	$0,044 \pm 0,001$
Naive Bayes	$0,154 \pm 0,007$	$0,098 \pm 0,005$	$0,092 \pm 0,009$	$0,113 \pm 0,007$	$0,073 \pm 0,002$
Linear SVM	$0,368 \pm 0,009$	$0,302 \pm 0,018$	$0,301 \pm 0,01$	$0,322 \pm 0,038$	$0,039 \pm 0,001$
Neural Net	$0,454 \pm 0,002$	$0,37 \pm 0,016$	$0,456 \pm 0,022$	$0,326 \pm 0,014$	$0,03 \pm 0,001$
RBF SVM	$0,322 \pm 0,013$	$0,095 \pm 0,013$	$0,387 \pm 0,027$	$0,067 \pm 0,009$	$0,04 \pm 0,001$
Adaboost	$0,445 \pm 0,012$	$0,325 \pm 0,008$	$0,442 \pm 0,014$	$0,274 \pm 0,008$	$0,032 \pm 0,001$
Random Forest	$0,387 \pm 0,009$	$0,245 \pm 0,018$	$0,436 \pm 0,024$	$0,184 \pm 0,015$	$0,034 \pm 0,001$

CUADRO 6.84. Resultados de Health con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,312 \pm 0,005$	$0,077 \pm 0,009$	$0,311 \pm 0,032$	$0,058 \pm 0,005$	$0,045 \pm 0,0$
Decision Tree	$0,66 \pm 0,009$	$0,447 \pm 0,031$	$0,455 \pm 0,035$	$0,449 \pm 0,027$	$0,021 \pm 0,0$
Naive Bayes	$0,168 \pm 0,014$	$0,101 \pm 0,004$	$0,096 \pm 0,008$	$0,114 \pm 0,003$	$0,072 \pm 0,002$
Linear SVM	$0,227 \pm 0,005$	$0,238 \pm 0,014$	$0,192 \pm 0,011$	$0,361 \pm 0,032$	$0,057 \pm 0,001$
Neural Net	$0,394 \pm 0,006$	$0,372 \pm 0,021$	$0,363 \pm 0,025$	$0,411 \pm 0,02$	$0,036 \pm 0,001$
RBF SVM	$0,304 \pm 0,007$	$0,037 \pm 0,007$	$0,212 \pm 0,042$	$0,035 \pm 0,004$	$0,047 \pm 0,001$
Adaboost	$0,636 \pm 0,007$	$0,473 \pm 0,037$	$0,582 \pm 0,054$	$0,417 \pm 0,025$	$0,02 \pm 0,001$
Random Forest	$0,645 \pm 0,006$	$0,336 \pm 0,01$	$0,466 \pm 0,036$	$0,282 \pm 0,008$	$0,02 \pm 0,0$

CUADRO 6.85. Resultados de Healthcon Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,255 \pm 0,042$	$0,066 \pm 0,005$	$0,29 \pm 0,019$	$0,049 \pm 0,002$	$0,047 \pm 0,0$
Decision Tree	$0,329 \pm 0,008$	$0,316 \pm 0,018$	$0,327 \pm 0,015$	$0,321 \pm 0,029$	$0,041 \pm 0,0$
Naive Bayes	$0,154 \pm 0,007$	$0,099 \pm 0,006$	$0,094 \pm 0,011$	$0,113 \pm 0,003$	$0,073 \pm 0,002$
Linear SVM	$0,297 \pm 0,016$	$0,258 \pm 0,013$	$0,236 \pm 0,013$	$0,306 \pm 0,021$	$0,047 \pm 0,002$
Neural Net	$0,395 \pm 0,013$	$0,335 \pm 0,012$	$0,371 \pm 0,019$	$0,318 \pm 0,017$	$0,036 \pm 0,001$
RBF SVM	$0,287 \pm 0,005$	$0,049 \pm 0,008$	$0,298 \pm 0,023$	$0,041 \pm 0,005$	$0,045 \pm 0,0$
Adaboost	$0,452 \pm 0,006$	$0,35 \pm 0,017$	$0,469 \pm 0,02$	$0,3 \pm 0,024$	$0,031 \pm 0,0$
Random Forest	$0,396 \pm 0,006$	$0,248 \pm 0,012$	$0,421 \pm 0,02$	$0,189 \pm 0,014$	$0,033 \pm 0,001$

CUADRO 6.86. Resultados de Health con Binary Relevance tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,274 \pm 0,046$	$0,076 \pm 0,004$	$0,302 \pm 0,029$	$0,055 \pm 0,004$	$0,044 \pm 0,0$
Decision Tree	$0,526 \pm 0,011$	$0,408 \pm 0,019$	$0,418 \pm 0,023$	$0,411 \pm 0,022$	$0,028 \pm 0,001$
Naive Bayes	$0,168 \pm 0,005$	$0,107 \pm 0,007$	$0,103 \pm 0,016$	$0,121 \pm 0,005$	$0,072 \pm 0,0$
Linear SVM	$0,212 \pm 0,007$	$0,234 \pm 0,011$	$0,189 \pm 0,01$	$0,355 \pm 0,023$	$0,059 \pm 0,001$
Neural Net	$0,341 \pm 0,013$	$0,324 \pm 0,022$	$0,31 \pm 0,025$	$0,369 \pm 0,025$	$0,041 \pm 0,001$
RBF SVM	$0,299 \pm 0,006$	$0,028 \pm 0,003$	$0,192 \pm 0,032$	$0,03 \pm 0,002$	$0,048 \pm 0,001$
Adaboost	$0,55 \pm 0,007$	$0,413 \pm 0,022$	$0,512 \pm 0,026$	$0,356 \pm 0,019$	$0,024 \pm 0,001$
Random Forest	$0,545 \pm 0,01$	$0,314 \pm 0,009$	$0,453 \pm 0,021$	$0,257 \pm 0,008$	$0,024 \pm 0,001$

CUADRO 6.87. Resultados de Health con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,33 \pm 0,007$	$0,072 \pm 0,009$	$0,21 \pm 0,043$	$0,062 \pm 0,007$	$0,05 \pm 0,001$
Decision Tree	$0,433 \pm 0,016$	$0,247 \pm 0,019$	$0,268 \pm 0,022$	$0,235 \pm 0,017$	$0,043 \pm 0,001$
Naive Bayes	$0,343 \pm 0,005$	$0,147 \pm 0,023$	$0,215 \pm 0,035$	$0,125 \pm 0,02$	$0,049 \pm 0,0$
Linear SVM	$0,409 \pm 0,012$	$0,219 \pm 0,019$	$0,197 \pm 0,019$	$0,28 \pm 0,018$	$0,061 \pm 0,002$
Neural Net	$0,455 \pm 0,039$	$0,272 \pm 0,051$	$0,308 \pm 0,071$	$0,262 \pm 0,043$	$0,042 \pm 0,006$
RBF SVM	$0,355 \pm 0,004$	$0,078 \pm 0,005$	$0,329 \pm 0,046$	$0,064 \pm 0,004$	$0,047 \pm 0,0$
Adaboost	$0,347 \pm 0,003$	$0,05 \pm 0,0$	$0,057 \pm 0,007$	$0,053 \pm 0,001$	$0,047 \pm 0,001$
Random Forest	$0,5 \pm 0,009$	$0,259 \pm 0,012$	$0,391 \pm 0,03$	$0,214 \pm 0,01$	$0,035 \pm 0,001$

CUADRO 6.88. Resultados de Health con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,148 \pm 0,01$	$0,16 \pm 0,023$	$0,222 \pm 0,04$	$0,174 \pm 0,021$	$0,077 \pm 0,003$
Decision Tree	$0,871 \pm 0,007$	$0,503 \pm 0,02$	$0,523 \pm 0,027$	$0,507 \pm 0,019$	$0,016 \pm 0,001$
Naive Bayes	$0,404 \pm 0,012$	$0,173 \pm 0,021$	$0,256 \pm 0,035$	$0,145 \pm 0,015$	$0,044 \pm 0,001$
Linear SVM	$0,585 \pm 0,011$	$0,285 \pm 0,021$	$0,272 \pm 0,025$	$0,327 \pm 0,022$	$0,041 \pm 0,001$
Neural Net	$0,607 \pm 0,014$	$0,365 \pm 0,028$	$0,394 \pm 0,032$	$0,372 \pm 0,02$	$0,032 \pm 0,002$
RBF SVM	$0,396 \pm 0,022$	$0,114 \pm 0,009$	$0,415 \pm 0,019$	$0,087 \pm 0,007$	$0,044 \pm 0,002$
Adaboost	$0,308 \pm 0,016$	$0,023 \pm 0,004$	$0,041 \pm 0,037$	$0,032 \pm 0,002$	$0,051 \pm 0,002$
Random Forest	$0,845 \pm 0,01$	$0,508 \pm 0,029$	$0,537 \pm 0,033$	$0,506 \pm 0,026$	$0,017 \pm 0,001$

CUADRO 6.89. Resultados de Health con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,313 \pm 0,03$	$0,092 \pm 0,008$	$0,26 \pm 0,046$	$0,077 \pm 0,004$	$0,052 \pm 0,003$
Decision Tree	$0,431 \pm 0,008$	$0,244 \pm 0,006$	$0,262 \pm 0,009$	$0,235 \pm 0,005$	$0,043 \pm 0,001$
Naive Bayes	$0,323 \pm 0,01$	$0,136 \pm 0,001$	$0,202 \pm 0,018$	$0,118 \pm 0,001$	$0,051 \pm 0,001$
Linear SVM	$0,352 \pm 0,005$	$0,192 \pm 0,008$	$0,169 \pm 0,009$	$0,256 \pm 0,008$	$0,063 \pm 0,001$
Neural Net	$0,439 \pm 0,017$	$0,271 \pm 0,013$	$0,306 \pm 0,014$	$0,258 \pm 0,015$	$0,043 \pm 0,002$
RBF SVM	$0,317 \pm 0,008$	$0,044 \pm 0,006$	$0,241 \pm 0,024$	$0,044 \pm 0,004$	$0,05 \pm 0,001$
Adaboost	$0,347 \pm 0,006$	$0,05 \pm 0,002$	$0,054 \pm 0,002$	$0,053 \pm 0,002$	$0,047 \pm 0,001$
Random Forest	$0,508 \pm 0,016$	$0,258 \pm 0,011$	$0,393 \pm 0,029$	$0,213 \pm 0,009$	$0,035 \pm 0,001$

CUADRO 6.90. Resultados de Health con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,214 \pm 0,014$	$0,142 \pm 0,02$	$0,224 \pm 0,025$	$0,158 \pm 0,026$	$0,071 \pm 0,002$
Decision Tree	$0,674 \pm 0,011$	$0,353 \pm 0,013$	$0,381 \pm 0,013$	$0,344 \pm 0,013$	$0,029 \pm 0,001$
Naive Bayes	$0,37 \pm 0,005$	$0,165 \pm 0,007$	$0,25 \pm 0,02$	$0,14 \pm 0,005$	$0,048 \pm 0,001$
Linear SVM	$0,501 \pm 0,011$	$0,266 \pm 0,029$	$0,245 \pm 0,03$	$0,321 \pm 0,031$	$0,049 \pm 0,002$
Neural Net	$0,542 \pm 0,018$	$0,323 \pm 0,015$	$0,354 \pm 0,019$	$0,327 \pm 0,018$	$0,039 \pm 0,001$
RBF SVM	$0,334 \pm 0,008$	$0,061 \pm 0,01$	$0,318 \pm 0,011$	$0,055 \pm 0,009$	$0,049 \pm 0,001$
Adaboost	$0,308 \pm 0,008$	$0,021 \pm 0,0$	$0,016 \pm 0,0$	$0,031 \pm 0,0$	$0,051 \pm 0,001$
Random Forest	$0,647 \pm 0,013$	$0,354 \pm 0,025$	$0,453 \pm 0,037$	$0,311 \pm 0,026$	$0,028 \pm 0,001$

CUADRO 6.91. Resultados de Health1 con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,105 \pm 0,007$	$0,151 \pm 0,012$	$0,625 \pm 0,044$	$0,104 \pm 0,01$	$0,069 \pm 0,002$
Decision Tree	$0,265 \pm 0,007$	$0,398 \pm 0,008$	$0,391 \pm 0,005$	$0,41 \pm 0,018$	$0,07 \pm 0,0$
Naive Bayes	$0,121 \pm 0,004$	$0,175 \pm 0,009$	$0,169 \pm 0,014$	$0,213 \pm 0,014$	$0,121 \pm 0,004$
Linear SVM	$0,35 \pm 0,008$	$0,447 \pm 0,014$	$0,455 \pm 0,019$	$0,448 \pm 0,018$	$0,06 \pm 0,001$
Neural Net	$0,439 \pm 0,012$	$0,554 \pm 0,01$	$0,677 \pm 0,019$	$0,482 \pm 0,009$	$0,046 \pm 0,001$
RBF SVM	$0,151 \pm 0,002$	$0,167 \pm 0,008$	$0,671 \pm 0,003$	$0,101 \pm 0,006$	$0,056 \pm 0,001$
Adaboost	$0,331 \pm 0,01$	$0,417 \pm 0,008$	$0,655 \pm 0,011$	$0,316 \pm 0,008$	$0,048 \pm 0,001$
Random Forest	$0,319 \pm 0,007$	$0,319 \pm 0,013$	$0,648 \pm 0,036$	$0,225 \pm 0,01$	$0,048 \pm 0,001$

CUADRO 6.92. Resultados de Recreation con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,118 \pm 0,005$	$0,202 \pm 0,008$	$0,755 \pm 0,04$	$0,124 \pm 0,008$	$0,059 \pm 0,001$
Decision Tree	$0,674 \pm 0,012$	$0,644 \pm 0,01$	$0,65 \pm 0,022$	$0,648 \pm 0,015$	$0,032 \pm 0,002$
Naive Bayes	$0,141 \pm 0,007$	$0,177 \pm 0,013$	$0,158 \pm 0,014$	$0,216 \pm 0,012$	$0,119 \pm 0,003$
Linear SVM	$0,33 \pm 0,008$	$0,454 \pm 0,015$	$0,397 \pm 0,014$	$0,551 \pm 0,019$	$0,067 \pm 0,001$
Neural Net	$0,476 \pm 0,007$	$0,603 \pm 0,018$	$0,628 \pm 0,022$	$0,593 \pm 0,017$	$0,043 \pm 0,001$
RBF SVM	$0,107 \pm 0,003$	$0,133 \pm 0,003$	$0,679 \pm 0,003$	$0,079 \pm 0,002$	$0,059 \pm 0,001$
Adaboost	$0,574 \pm 0,009$	$0,625 \pm 0,009$	$0,816 \pm 0,015$	$0,515 \pm 0,01$	$0,033 \pm 0,001$
Random Forest	$0,625 \pm 0,008$	$0,543 \pm 0,016$	$0,787 \pm 0,033$	$0,44 \pm 0,013$	$0,031 \pm 0,001$

CUADRO 6.93. Resultados de Recreation con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,11 \pm 0,008$	$0,164 \pm 0,019$	$0,642 \pm 0,037$	$0,101 \pm 0,014$	$0,062 \pm 0,001$
Decision Tree	$0,29 \pm 0,013$	$0,426 \pm 0,011$	$0,43 \pm 0,008$	$0,428 \pm 0,014$	$0,065 \pm 0,001$
Naive Bayes	$0,12 \pm 0,003$	$0,174 \pm 0,009$	$0,166 \pm 0,01$	$0,211 \pm 0,01$	$0,121 \pm 0,001$
Linear SVM	$0,285 \pm 0,004$	$0,396 \pm 0,016$	$0,37 \pm 0,011$	$0,435 \pm 0,025$	$0,071 \pm 0,001$
Neural Net	$0,372 \pm 0,007$	$0,495 \pm 0,008$	$0,522 \pm 0,01$	$0,481 \pm 0,004$	$0,055 \pm 0,001$
RBF SVM	$0,062 \pm 0,001$	$0,103 \pm 0,007$	$0,666 \pm 0,019$	$0,059 \pm 0,005$	$0,061 \pm 0,001$
Adaboost	$0,34 \pm 0,014$	$0,431 \pm 0,023$	$0,684 \pm 0,033$	$0,324 \pm 0,019$	$0,047 \pm 0,001$
Random Forest	$0,343 \pm 0,012$	$0,35 \pm 0,015$	$0,685 \pm 0,028$	$0,252 \pm 0,013$	$0,047 \pm 0,001$

CUADRO 6.94. Resultados de Recreation con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,143 \pm 0,006$	$0,2 \pm 0,004$	$0,709 \pm 0,057$	$0,125 \pm 0,003$	$0,059 \pm 0,0$
Decision Tree	$0,485 \pm 0,011$	$0,55 \pm 0,015$	$0,547 \pm 0,019$	$0,561 \pm 0,01$	$0,046 \pm 0,001$
Naive Bayes	$0,139 \pm 0,007$	$0,177 \pm 0,009$	$0,162 \pm 0,019$	$0,216 \pm 0,005$	$0,12 \pm 0,002$
Linear SVM	$0,281 \pm 0,009$	$0,424 \pm 0,014$	$0,364 \pm 0,009$	$0,528 \pm 0,027$	$0,074 \pm 0,001$
Neural Net	$0,414 \pm 0,009$	$0,539 \pm 0,013$	$0,525 \pm 0,013$	$0,569 \pm 0,017$	$0,051 \pm 0,001$
RBF SVM	$0,054 \pm 0,003$	$0,09 \pm 0,008$	$0,673 \pm 0,018$	$0,051 \pm 0,006$	$0,061 \pm 0,0$
Adaboost	$0,464 \pm 0,007$	$0,549 \pm 0,014$	$0,769 \pm 0,024$	$0,438 \pm 0,012$	$0,039 \pm 0,001$
Random Forest	$0,457 \pm 0,008$	$0,46 \pm 0,012$	$0,752 \pm 0,031$	$0,352 \pm 0,011$	$0,039 \pm 0,001$

CUADRO 6.95. Resultados de Recreation con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,164 \pm 0,017$	$0,183 \pm 0,013$	$0,474 \pm 0,029$	$0,151 \pm 0,011$	$0,088 \pm 0,001$
Decision Tree	$0,392 \pm 0,006$	$0,356 \pm 0,013$	$0,385 \pm 0,016$	$0,34 \pm 0,011$	$0,068 \pm 0,001$
Naive Bayes	$0,337 \pm 0,012$	$0,269 \pm 0,016$	$0,403 \pm 0,053$	$0,227 \pm 0,01$	$0,072 \pm 0,001$
Linear SVM	$0,399 \pm 0,009$	$0,323 \pm 0,01$	$0,287 \pm 0,009$	$0,402 \pm 0,013$	$0,093 \pm 0,003$
Neural Net	$0,427 \pm 0,009$	$0,412 \pm 0,012$	$0,455 \pm 0,024$	$0,399 \pm 0,009$	$0,065 \pm 0,001$
RBF SVM	$0,379 \pm 0,014$	$0,22 \pm 0,004$	$0,635 \pm 0,022$	$0,185 \pm 0,003$	$0,067 \pm 0,002$
Adaboost	$0,177 \pm 0,009$	$0,04 \pm 0,001$	$0,047 \pm 0,001$	$0,063 \pm 0,001$	$0,088 \pm 0,001$
Random Forest	$0,469 \pm 0,01$	$0,371 \pm 0,01$	$0,527 \pm 0,049$	$0,318 \pm 0,006$	$0,057 \pm 0,001$

CUADRO 6.96. Resultados de Recreation con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,163 \pm 0,011$	$0,199 \pm 0,017$	$0,331 \pm 0,028$	$0,257 \pm 0,018$	$0,118 \pm 0,007$
Decision Tree	$0,839 \pm 0,006$	$0,666 \pm 0,017$	$0,677 \pm 0,025$	$0,667 \pm 0,015$	$0,029 \pm 0,001$
Naive Bayes	$0,414 \pm 0,006$	$0,332 \pm 0,013$	$0,474 \pm 0,032$	$0,283 \pm 0,01$	$0,065 \pm 0,001$
Linear SVM	$0,557 \pm 0,005$	$0,427 \pm 0,01$	$0,398 \pm 0,011$	$0,482 \pm 0,008$	$0,068 \pm 0,002$
Neural Net	$0,575 \pm 0,015$	$0,517 \pm 0,034$	$0,572 \pm 0,039$	$0,499 \pm 0,034$	$0,051 \pm 0,003$
RBF SVM	$0,429 \pm 0,009$	$0,253 \pm 0,009$	$0,672 \pm 0,02$	$0,206 \pm 0,006$	$0,063 \pm 0,001$
Adaboost	$0,192 \pm 0,01$	$0,04 \pm 0,001$	$0,06 \pm 0,009$	$0,062 \pm 0,001$	$0,087 \pm 0,001$
Random Forest	$0,815 \pm 0,005$	$0,667 \pm 0,015$	$0,689 \pm 0,016$	$0,663 \pm 0,015$	$0,03 \pm 0,001$

CUADRO 6.97. Resultados de Recreation con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,229 \pm 0,007$	$0,22 \pm 0,013$	$0,404 \pm 0,02$	$0,186 \pm 0,016$	$0,084 \pm 0,001$
Decision Tree	$0,401 \pm 0,008$	$0,375 \pm 0,016$	$0,407 \pm 0,015$	$0,356 \pm 0,016$	$0,067 \pm 0,002$
Naive Bayes	$0,322 \pm 0,008$	$0,266 \pm 0,019$	$0,387 \pm 0,033$	$0,226 \pm 0,015$	$0,074 \pm 0,001$
Linear SVM	$0,354 \pm 0,006$	$0,305 \pm 0,01$	$0,275 \pm 0,012$	$0,373 \pm 0,013$	$0,096 \pm 0,003$
Neural Net	$0,388 \pm 0,022$	$0,382 \pm 0,019$	$0,429 \pm 0,024$	$0,366 \pm 0,016$	$0,071 \pm 0,003$
RBF SVM	$0,274 \pm 0,012$	$0,157 \pm 0,01$	$0,603 \pm 0,02$	$0,135 \pm 0,008$	$0,078 \pm 0,001$
Adaboost	$0,177 \pm 0,004$	$0,04 \pm 0,001$	$0,047 \pm 0,001$	$0,063 \pm 0,001$	$0,088 \pm 0,001$
Random Forest	$0,481 \pm 0,004$	$0,394 \pm 0,017$	$0,565 \pm 0,047$	$0,336 \pm 0,014$	$0,055 \pm 0,001$

CUADRO 6.98. Resultados de Recreation con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,194 \pm 0,011$	$0,206 \pm 0,007$	$0,336 \pm 0,011$	$0,213 \pm 0,006$	$0,102 \pm 0,002$
Decision Tree	$0,699 \pm 0,005$	$0,52 \pm 0,013$	$0,54 \pm 0,013$	$0,51 \pm 0,017$	$0,044 \pm 0,001$
Naive Bayes	$0,378 \pm 0,004$	$0,306 \pm 0,01$	$0,439 \pm 0,035$	$0,261 \pm 0,008$	$0,068 \pm 0,001$
Linear SVM	$0,483 \pm 0,009$	$0,384 \pm 0,012$	$0,351 \pm 0,014$	$0,444 \pm 0,019$	$0,077 \pm 0,002$
Neural Net	$0,517 \pm 0,007$	$0,471 \pm 0,024$	$0,505 \pm 0,031$	$0,467 \pm 0,02$	$0,059 \pm 0,002$
RBF SVM	$0,325 \pm 0,01$	$0,176 \pm 0,014$	$0,646 \pm 0,032$	$0,151 \pm 0,009$	$0,074 \pm 0,002$
Adaboost	$0,178 \pm 0,006$	$0,04 \pm 0,001$	$0,052 \pm 0,001$	$0,062 \pm 0,0$	$0,088 \pm 0,001$
Random Forest	$0,633 \pm 0,011$	$0,517 \pm 0,016$	$0,621 \pm 0,014$	$0,463 \pm 0,021$	$0,044 \pm 0,002$

CUADRO 6.99. Resultados de Recreation con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,34 \pm 0,024$	$0,034 \pm 0,002$	$0,2 \pm 0,024$	$0,034 \pm 0,001$	$0,035 \pm 0,0$
Decision Tree	$0,334 \pm 0,011$	$0,216 \pm 0,01$	$0,215 \pm 0,008$	$0,226 \pm 0,018$	$0,037 \pm 0,0$
Naive Bayes	$0,18 \pm 0,008$	$0,072 \pm 0,008$	$0,081 \pm 0,014$	$0,072 \pm 0,007$	$0,051 \pm 0,001$
Linear SVM	$0,389 \pm 0,005$	$0,247 \pm 0,016$	$0,244 \pm 0,018$	$0,276 \pm 0,024$	$0,032 \pm 0,0$
Neural Net	$0,508 \pm 0,02$	$0,335 \pm 0,017$	$0,446 \pm 0,035$	$0,287 \pm 0,021$	$0,023 \pm 0,001$
RBF SVM	$0,326 \pm 0,009$	$0,032 \pm 0,001$	$0,182 \pm 0,035$	$0,028 \pm 0,001$	$0,029 \pm 0,0$
Adaboost	$0,395 \pm 0,012$	$0,252 \pm 0,014$	$0,378 \pm 0,011$	$0,198 \pm 0,015$	$0,025 \pm 0,001$
Random Forest	$0,361 \pm 0,019$	$0,117 \pm 0,011$	$0,333 \pm 0,039$	$0,078 \pm 0,007$	$0,026 \pm 0,001$

CUADRO 6.100. Resultados de Reference con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,057 \pm 0,006$	$0,042 \pm 0,013$	$0,309 \pm 0,057$	$0,024 \pm 0,008$	$0,033 \pm 0,0$
Decision Tree	$0,664 \pm 0,002$	$0,391 \pm 0,026$	$0,413 \pm 0,035$	$0,387 \pm 0,023$	$0,017 \pm 0,0$
Naive Bayes	$0,192 \pm 0,009$	$0,068 \pm 0,004$	$0,077 \pm 0,015$	$0,071 \pm 0,005$	$0,051 \pm 0,002$
Linear SVM	$0,369 \pm 0,006$	$0,255 \pm 0,007$	$0,206 \pm 0,007$	$0,39 \pm 0,009$	$0,039 \pm 0,0$
Neural Net	$0,549 \pm 0,011$	$0,374 \pm 0,021$	$0,428 \pm 0,042$	$0,362 \pm 0,021$	$0,021 \pm 0,001$
RBF SVM	$0,172 \pm 0,008$	$0,023 \pm 0,001$	$0,151 \pm 0,0$	$0,015 \pm 0,0$	$0,03 \pm 0,0$
Adaboost	$0,686 \pm 0,011$	$0,448 \pm 0,043$	$0,58 \pm 0,056$	$0,381 \pm 0,035$	$0,014 \pm 0,0$
Random Forest	$0,607 \pm 0,01$	$0,232 \pm 0,015$	$0,437 \pm 0,036$	$0,176 \pm 0,011$	$0,016 \pm 0,0$

CUADRO 6.101. Resultados de Reference con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,35 \pm 0,019$	$0,038 \pm 0,003$	$0,216 \pm 0,04$	$0,037 \pm 0,002$	$0,035 \pm 0,001$
Decision Tree	$0,357 \pm 0,018$	$0,238 \pm 0,015$	$0,246 \pm 0,024$	$0,242 \pm 0,016$	$0,035 \pm 0,001$
Naive Bayes	$0,188 \pm 0,008$	$0,068 \pm 0,009$	$0,076 \pm 0,017$	$0,071 \pm 0,008$	$0,052 \pm 0,001$
Linear SVM	$0,304 \pm 0,011$	$0,207 \pm 0,016$	$0,178 \pm 0,011$	$0,285 \pm 0,036$	$0,042 \pm 0,001$
Neural Net	$0,416 \pm 0,013$	$0,285 \pm 0,009$	$0,311 \pm 0,007$	$0,279 \pm 0,017$	$0,029 \pm 0,001$
RBF SVM	$0,331 \pm 0,009$	$0,027 \pm 0,001$	$0,115 \pm 0,035$	$0,028 \pm 0,001$	$0,032 \pm 0,0$
Adaboost	$0,396 \pm 0,01$	$0,257 \pm 0,012$	$0,405 \pm 0,03$	$0,199 \pm 0,01$	$0,025 \pm 0,001$
Random Forest	$0,364 \pm 0,013$	$0,122 \pm 0,008$	$0,313 \pm 0,04$	$0,083 \pm 0,005$	$0,026 \pm 0,001$

CUADRO 6.102. Resultados de Reference con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,102 \pm 0,012$	$0,045 \pm 0,007$	$0,306 \pm 0,02$	$0,026 \pm 0,005$	$0,032 \pm 0,001$
Decision Tree	$0,515 \pm 0,01$	$0,323 \pm 0,012$	$0,319 \pm 0,008$	$0,345 \pm 0,024$	$0,025 \pm 0,001$
Naive Bayes	$0,201 \pm 0,006$	$0,074 \pm 0,011$	$0,081 \pm 0,02$	$0,076 \pm 0,008$	$0,051 \pm 0,002$
Linear SVM	$0,315 \pm 0,011$	$0,226 \pm 0,01$	$0,179 \pm 0,005$	$0,359 \pm 0,03$	$0,044 \pm 0,001$
Neural Net	$0,453 \pm 0,016$	$0,358 \pm 0,008$	$0,373 \pm 0,028$	$0,372 \pm 0,015$	$0,027 \pm 0,001$
RBF SVM	$0,122 \pm 0,01$	$0,017 \pm 0,002$	$0,12 \pm 0,043$	$0,011 \pm 0,001$	$0,031 \pm 0,001$
Adaboost	$0,569 \pm 0,01$	$0,372 \pm 0,01$	$0,518 \pm 0,027$	$0,317 \pm 0,009$	$0,018 \pm 0,001$
Random Forest	$0,494 \pm 0,004$	$0,182 \pm 0,004$	$0,372 \pm 0,027$	$0,133 \pm 0,003$	$0,02 \pm 0,0$

CUADRO 6.103. Resultados de Reference con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,382 \pm 0,006$	$0,051 \pm 0,005$	$0,179 \pm 0,022$	$0,048 \pm 0,003$	$0,037 \pm 0,0$
Decision Tree	$0,431 \pm 0,015$	$0,161 \pm 0,02$	$0,174 \pm 0,027$	$0,155 \pm 0,018$	$0,036 \pm 0,001$
Naive Bayes	$0,371 \pm 0,007$	$0,119 \pm 0,005$	$0,179 \pm 0,015$	$0,103 \pm 0,004$	$0,038 \pm 0,001$
Linear SVM	$0,431 \pm 0,011$	$0,184 \pm 0,017$	$0,174 \pm 0,024$	$0,237 \pm 0,025$	$0,05 \pm 0,002$
Neural Net	$0,456 \pm 0,04$	$0,181 \pm 0,022$	$0,186 \pm 0,028$	$0,192 \pm 0,022$	$0,038 \pm 0,005$
RBF SVM	$0,388 \pm 0,007$	$0,032 \pm 0,002$	$0,195 \pm 0,01$	$0,037 \pm 0,001$	$0,037 \pm 0,0$
Adaboost	$0,378 \pm 0,006$	$0,019 \pm 0,0$	$0,014 \pm 0,0$	$0,03 \pm 0,0$	$0,037 \pm 0,0$
Random Forest	$0,493 \pm 0,008$	$0,15 \pm 0,01$	$0,224 \pm 0,023$	$0,126 \pm 0,007$	$0,031 \pm 0,001$

CUADRO 6.104. Resultados de Reference con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,1 \pm 0,008$	$0,091 \pm 0,018$	$0,21 \pm 0,027$	$0,101 \pm 0,018$	$0,063 \pm 0,003$
Decision Tree	$0,873 \pm 0,005$	$0,45 \pm 0,024$	$0,462 \pm 0,027$	$0,452 \pm 0,025$	$0,012 \pm 0,0$
Naive Bayes	$0,432 \pm 0,007$	$0,173 \pm 0,03$	$0,253 \pm 0,029$	$0,153 \pm 0,03$	$0,035 \pm 0,001$
Linear SVM	$0,603 \pm 0,008$	$0,27 \pm 0,012$	$0,256 \pm 0,012$	$0,325 \pm 0,034$	$0,033 \pm 0,001$
Neural Net	$0,621 \pm 0,013$	$0,305 \pm 0,023$	$0,333 \pm 0,037$	$0,313 \pm 0,038$	$0,027 \pm 0,002$
RBF SVM	$0,404 \pm 0,008$	$0,042 \pm 0,001$	$0,226 \pm 0,031$	$0,042 \pm 0,001$	$0,036 \pm 0,0$
Adaboost	$0,378 \pm 0,007$	$0,019 \pm 0,0$	$0,014 \pm 0,0$	$0,03 \pm 0,0$	$0,037 \pm 0,0$
Random Forest	$0,837 \pm 0,003$	$0,451 \pm 0,023$	$0,475 \pm 0,024$	$0,46 \pm 0,023$	$0,014 \pm 0,0$

CUADRO 6.105. Resultados de Reference con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,383 \pm 0,011$	$0,067 \pm 0,006$	$0,194 \pm 0,016$	$0,057 \pm 0,004$	$0,037 \pm 0,001$
Decision Tree	$0,442 \pm 0,016$	$0,169 \pm 0,014$	$0,186 \pm 0,023$	$0,161 \pm 0,009$	$0,036 \pm 0,001$
Naive Bayes	$0,359 \pm 0,009$	$0,12 \pm 0,016$	$0,183 \pm 0,034$	$0,105 \pm 0,012$	$0,039 \pm 0,001$
Linear SVM	$0,38 \pm 0,007$	$0,169 \pm 0,01$	$0,15 \pm 0,009$	$0,225 \pm 0,019$	$0,051 \pm 0,002$
Neural Net	$0,428 \pm 0,014$	$0,185 \pm 0,026$	$0,208 \pm 0,036$	$0,181 \pm 0,023$	$0,037 \pm 0,003$
RBF SVM	$0,383 \pm 0,011$	$0,026 \pm 0,001$	$0,12 \pm 0,024$	$0,034 \pm 0,001$	$0,037 \pm 0,001$
Adaboost	$0,378 \pm 0,011$	$0,019 \pm 0,0$	$0,014 \pm 0,0$	$0,03 \pm 0,0$	$0,037 \pm 0,001$
Random Forest	$0,502 \pm 0,009$	$0,153 \pm 0,011$	$0,242 \pm 0,035$	$0,129 \pm 0,007$	$0,03 \pm 0,001$

CUADRO 6.106. Resultados de Reference con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,179 \pm 0,01$	$0,093 \pm 0,014$	$0,178 \pm 0,041$	$0,097 \pm 0,013$	$0,053 \pm 0,001$
Decision Tree	$0,721 \pm 0,012$	$0,325 \pm 0,019$	$0,34 \pm 0,021$	$0,323 \pm 0,024$	$0,021 \pm 0,001$
Naive Bayes	$0,408 \pm 0,007$	$0,151 \pm 0,008$	$0,221 \pm 0,017$	$0,132 \pm 0,007$	$0,037 \pm 0,001$
Linear SVM	$0,519 \pm 0,016$	$0,23 \pm 0,008$	$0,215 \pm 0,008$	$0,267 \pm 0,014$	$0,038 \pm 0,002$
Neural Net	$0,537 \pm 0,028$	$0,251 \pm 0,02$	$0,251 \pm 0,028$	$0,275 \pm 0,021$	$0,034 \pm 0,002$
RBF SVM	$0,386 \pm 0,014$	$0,028 \pm 0,001$	$0,178 \pm 0,015$	$0,035 \pm 0,001$	$0,037 \pm 0,001$
Adaboost	$0,379 \pm 0,013$	$0,022 \pm 0,006$	$0,02 \pm 0,012$	$0,032 \pm 0,004$	$0,037 \pm 0,001$
Random Forest	$0,612 \pm 0,023$	$0,243 \pm 0,01$	$0,351 \pm 0,035$	$0,21 \pm 0,009$	$0,025 \pm 0,001$

CUADRO 6.107. Resultados de Reference con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,079 \pm 0,022$	$0,027 \pm 0,003$	$0,173 \pm 0,014$	$0,018 \pm 0,002$	$0,036 \pm 0,001$
Decision Tree	$0,221 \pm 0,007$	$0,255 \pm 0,025$	$0,253 \pm 0,028$	$0,282 \pm 0,027$	$0,045 \pm 0,001$
Naive Bayes	$0,102 \pm 0,009$	$0,072 \pm 0,006$	$0,094 \pm 0,019$	$0,066 \pm 0,004$	$0,053 \pm 0,001$
Linear SVM	$0,29 \pm 0,007$	$0,286 \pm 0,013$	$0,295 \pm 0,014$	$0,298 \pm 0,02$	$0,037 \pm 0,001$
Neural Net	$0,415 \pm 0,01$	$0,383 \pm 0,014$	$0,494 \pm 0,018$	$0,335 \pm 0,014$	$0,027 \pm 0,001$
RBF SVM	$0,033 \pm 0,005$	$0,01 \pm 0,001$	$0,124 \pm 0,015$	$0,005 \pm 0,001$	$0,035 \pm 0,0$
Adaboost	$0,273 \pm 0,004$	$0,278 \pm 0,017$	$0,45 \pm 0,024$	$0,213 \pm 0,015$	$0,029 \pm 0,0$
Random Forest	$0,192 \pm 0,012$	$0,106 \pm 0,01$	$0,387 \pm 0,037$	$0,067 \pm 0,006$	$0,031 \pm 0,001$

CUADRO 6.108. Resultados de Science con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,075 \pm 0,014$	$0,059 \pm 0,007$	$0,304 \pm 0,043$	$0,041 \pm 0,006$	$0,037 \pm 0,002$
Decision Tree	$0,533 \pm 0,013$	$0,48 \pm 0,015$	$0,493 \pm 0,024$	$0,488 \pm 0,017$	$0,025 \pm 0,001$
Naive Bayes	$0,132 \pm 0,009$	$0,084 \pm 0,007$	$0,106 \pm 0,015$	$0,077 \pm 0,006$	$0,052 \pm 0,001$
Linear SVM	$0,245 \pm 0,011$	$0,293 \pm 0,011$	$0,237 \pm 0,009$	$0,443 \pm 0,026$	$0,049 \pm 0,001$
Neural Net	$0,385 \pm 0,009$	$0,461 \pm 0,014$	$0,495 \pm 0,021$	$0,471 \pm 0,022$	$0,029 \pm 0,001$
RBF SVM	$0,02 \pm 0,002$	$0,005 \pm 0,0$	$0,095 \pm 0,029$	$0,003 \pm 0,0$	$0,036 \pm 0,0$
Adaboost	$0,531 \pm 0,012$	$0,587 \pm 0,019$	$0,763 \pm 0,021$	$0,501 \pm 0,019$	$0,02 \pm 0,001$
Random Forest	$0,466 \pm 0,017$	$0,274 \pm 0,01$	$0,488 \pm 0,019$	$0,21 \pm 0,006$	$0,022 \pm 0,001$

CUADRO 6.109. Resultados de Science con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,199 \pm 0,018$	$0,065 \pm 0,015$	$0,168 \pm 0,045$	$0,059 \pm 0,012$	$0,049 \pm 0,002$
Decision Tree	$0,322 \pm 0,006$	$0,206 \pm 0,009$	$0,224 \pm 0,009$	$0,202 \pm 0,011$	$0,045 \pm 0,001$
Naive Bayes	$0,281 \pm 0,006$	$0,12 \pm 0,013$	$0,244 \pm 0,038$	$0,097 \pm 0,009$	$0,044 \pm 0,0$
Linear SVM	$0,342 \pm 0,013$	$0,222 \pm 0,016$	$0,214 \pm 0,012$	$0,261 \pm 0,022$	$0,057 \pm 0,002$
Neural Net	$0,37 \pm 0,039$	$0,215 \pm 0,043$	$0,225 \pm 0,055$	$0,23 \pm 0,043$	$0,046 \pm 0,005$
RBF SVM	$0,204 \pm 0,012$	$0,02 \pm 0,002$	$0,139 \pm 0,022$	$0,031 \pm 0,001$	$0,048 \pm 0,001$
Adaboost	$0,217 \pm 0,01$	$0,024 \pm 0,001$	$0,032 \pm 0,01$	$0,035 \pm 0,001$	$0,048 \pm 0,001$
Random Forest	$0,371 \pm 0,006$	$0,17 \pm 0,007$	$0,285 \pm 0,032$	$0,144 \pm 0,005$	$0,038 \pm 0,001$

CUADRO 6.110. Resultados de Science con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,117 \pm 0,007$	$0,11 \pm 0,018$	$0,228 \pm 0,022$	$0,128 \pm 0,025$	$0,071 \pm 0,006$
Decision Tree	$0,776 \pm 0,01$	$0,536 \pm 0,038$	$0,556 \pm 0,041$	$0,54 \pm 0,036$	$0,022 \pm 0,001$
Naive Bayes	$0,344 \pm 0,014$	$0,165 \pm 0,005$	$0,317 \pm 0,031$	$0,134 \pm 0,003$	$0,04 \pm 0,001$
Linear SVM	$0,526 \pm 0,007$	$0,325 \pm 0,01$	$0,322 \pm 0,023$	$0,356 \pm 0,017$	$0,04 \pm 0,001$
Neural Net	$0,508 \pm 0,023$	$0,313 \pm 0,018$	$0,336 \pm 0,016$	$0,343 \pm 0,022$	$0,038 \pm 0,003$
RBF SVM	$0,235 \pm 0,011$	$0,038 \pm 0,002$	$0,23 \pm 0,027$	$0,041 \pm 0,001$	$0,047 \pm 0,001$
Adaboost	$0,181 \pm 0,007$	$0,023 \pm 0,008$	$0,036 \pm 0,019$	$0,036 \pm 0,008$	$0,048 \pm 0,001$
Random Forest	$0,767 \pm 0,006$	$0,537 \pm 0,012$	$0,604 \pm 0,016$	$0,513 \pm 0,016$	$0,02 \pm 0,001$

CUADRO 6.111. Resultados de Science con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,348 \pm 0,012$	$0,05 \pm 0,006$	$0,197 \pm 0,029$	$0,041 \pm 0,004$	$0,029 \pm 0,001$
Decision Tree	$0,423 \pm 0,013$	$0,263 \pm 0,011$	$0,255 \pm 0,009$	$0,283 \pm 0,014$	$0,029 \pm 0,001$
Naive Bayes	$0,227 \pm 0,015$	$0,069 \pm 0,005$	$0,062 \pm 0,005$	$0,084 \pm 0,007$	$0,049 \pm 0,001$
Linear SVM	$0,462 \pm 0,008$	$0,266 \pm 0,018$	$0,259 \pm 0,016$	$0,287 \pm 0,025$	$0,027 \pm 0,001$
Neural Net	$0,557 \pm 0,01$	$0,385 \pm 0,019$	$0,523 \pm 0,032$	$0,329 \pm 0,021$	$0,019 \pm 0,0$
RBF SVM	$0,313 \pm 0,012$	$0,047 \pm 0,002$	$0,125 \pm 0,002$	$0,033 \pm 0,001$	$0,024 \pm 0,001$
Adaboost	$0,493 \pm 0,013$	$0,3 \pm 0,007$	$0,449 \pm 0,02$	$0,238 \pm 0,007$	$0,02 \pm 0,001$
Random Forest	$0,461 \pm 0,015$	$0,114 \pm 0,012$	$0,321 \pm 0,03$	$0,082 \pm 0,01$	$0,021 \pm 0,001$

CUADRO 6.112. Resultados de Social con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,184 \pm 0,01$	$0,081 \pm 0,011$	$0,37 \pm 0,038$	$0,049 \pm 0,007$	$0,027 \pm 0,0$
Decision Tree	$0,744 \pm 0,006$	$0,491 \pm 0,017$	$0,507 \pm 0,023$	$0,489 \pm 0,015$	$0,012 \pm 0,0$
Naive Bayes	$0,239 \pm 0,007$	$0,072 \pm 0,007$	$0,065 \pm 0,007$	$0,088 \pm 0,007$	$0,049 \pm 0,001$
Linear SVM	$0,344 \pm 0,007$	$0,254 \pm 0,009$	$0,196 \pm 0,007$	$0,417 \pm 0,02$	$0,038 \pm 0,001$
Neural Net	$0,53 \pm 0,007$	$0,396 \pm 0,016$	$0,395 \pm 0,018$	$0,43 \pm 0,019$	$0,021 \pm 0,001$
RBF SVM	$0,12 \pm 0,003$	$0,029 \pm 0,002$	$0,128 \pm 0,0$	$0,017 \pm 0,001$	$0,029 \pm 0,0$
Adaboost	$0,734 \pm 0,007$	$0,536 \pm 0,018$	$0,713 \pm 0,036$	$0,451 \pm 0,02$	$0,011 \pm 0,0$
Random Forest	$0,711 \pm 0,007$	$0,266 \pm 0,006$	$0,52 \pm 0,02$	$0,206 \pm 0,005$	$0,013 \pm 0,0$

CUADRO 6.113. Resultados de Social con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,245 \pm 0,007$	$0,059 \pm 0,007$	$0,219 \pm 0,05$	$0,04 \pm 0,004$	$0,03 \pm 0,0$
Decision Tree	$0,431 \pm 0,013$	$0,29 \pm 0,014$	$0,293 \pm 0,02$	$0,299 \pm 0,008$	$0,028 \pm 0,001$
Naive Bayes	$0,228 \pm 0,01$	$0,075 \pm 0,01$	$0,074 \pm 0,019$	$0,089 \pm 0,008$	$0,049 \pm 0,001$
Linear SVM	$0,366 \pm 0,003$	$0,213 \pm 0,007$	$0,181 \pm 0,007$	$0,291 \pm 0,018$	$0,036 \pm 0,001$
Neural Net	$0,461 \pm 0,006$	$0,313 \pm 0,004$	$0,326 \pm 0,014$	$0,318 \pm 0,01$	$0,026 \pm 0,0$
RBF SVM	$0,173 \pm 0,004$	$0,026 \pm 0,001$	$0,121 \pm 0,01$	$0,016 \pm 0,001$	$0,028 \pm 0,0$
Adaboost	$0,5 \pm 0,009$	$0,323 \pm 0,017$	$0,47 \pm 0,038$	$0,26 \pm 0,013$	$0,02 \pm 0,0$
Random Forest	$0,466 \pm 0,01$	$0,128 \pm 0,01$	$0,355 \pm 0,025$	$0,091 \pm 0,007$	$0,021 \pm 0,0$

CUADRO 6.114. Resultados de Social con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,238 \pm 0,011$	$0,078 \pm 0,008$	$0,333 \pm 0,026$	$0,049 \pm 0,006$	$0,027 \pm 0,0$
Decision Tree	$0,596 \pm 0,008$	$0,391 \pm 0,019$	$0,396 \pm 0,021$	$0,402 \pm 0,023$	$0,019 \pm 0,001$
Naive Bayes	$0,241 \pm 0,005$	$0,075 \pm 0,004$	$0,072 \pm 0,013$	$0,09 \pm 0,005$	$0,05 \pm 0,002$
Linear SVM	$0,294 \pm 0,011$	$0,221 \pm 0,008$	$0,167 \pm 0,006$	$0,382 \pm 0,034$	$0,042 \pm 0,001$
Neural Net	$0,434 \pm 0,013$	$0,33 \pm 0,01$	$0,297 \pm 0,012$	$0,408 \pm 0,012$	$0,028 \pm 0,001$
RBF SVM	$0,052 \pm 0,005$	$0,017 \pm 0,003$	$0,128 \pm 0,0$	$0,01 \pm 0,002$	$0,031 \pm 0,0$
Adaboost	$0,624 \pm 0,008$	$0,444 \pm 0,021$	$0,603 \pm 0,021$	$0,371 \pm 0,021$	$0,015 \pm 0,0$
Random Forest	$0,603 \pm 0,007$	$0,21 \pm 0,012$	$0,444 \pm 0,038$	$0,156 \pm 0,008$	$0,016 \pm 0,0$

CUADRO 6.115. Resultados de Social con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,423 \pm 0,006$	$0,073 \pm 0,007$	$0,163 \pm 0,02$	$0,064 \pm 0,005$	$0,032 \pm 0,0$
Decision Tree	$0,517 \pm 0,011$	$0,197 \pm 0,011$	$0,218 \pm 0,018$	$0,186 \pm 0,008$	$0,028 \pm 0,001$
Naive Bayes	$0,399 \pm 0,006$	$0,085 \pm 0,007$	$0,149 \pm 0,034$	$0,077 \pm 0,004$	$0,034 \pm 0,0$
Linear SVM	$0,505 \pm 0,01$	$0,2 \pm 0,004$	$0,198 \pm 0,015$	$0,244 \pm 0,013$	$0,039 \pm 0,002$
Neural Net	$0,548 \pm 0,024$	$0,225 \pm 0,046$	$0,249 \pm 0,063$	$0,22 \pm 0,038$	$0,028 \pm 0,003$
RBF SVM	$0,489 \pm 0,005$	$0,055 \pm 0,002$	$0,112 \pm 0,01$	$0,052 \pm 0,001$	$0,029 \pm 0,0$
Adaboost	$0,45 \pm 0,002$	$0,034 \pm 0,0$	$0,037 \pm 0,0$	$0,038 \pm 0,0$	$0,03 \pm 0,0$
Random Forest	$0,573 \pm 0,004$	$0,142 \pm 0,014$	$0,331 \pm 0,046$	$0,114 \pm 0,008$	$0,024 \pm 0,0$

CUADRO 6.116. Resultados de Social con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,211 \pm 0,009$	$0,138 \pm 0,016$	$0,181 \pm 0,036$	$0,159 \pm 0,02$	$0,053 \pm 0,002$
Decision Tree	$0,886 \pm 0,003$	$0,529 \pm 0,018$	$0,54 \pm 0,024$	$0,539 \pm 0,017$	$0,011 \pm 0,0$
Naive Bayes	$0,448 \pm 0,007$	$0,118 \pm 0,008$	$0,225 \pm 0,042$	$0,102 \pm 0,004$	$0,031 \pm 0,0$
Linear SVM	$0,638 \pm 0,012$	$0,287 \pm 0,022$	$0,286 \pm 0,019$	$0,314 \pm 0,029$	$0,027 \pm 0,001$
Neural Net	$0,651 \pm 0,023$	$0,338 \pm 0,027$	$0,393 \pm 0,042$	$0,328 \pm 0,032$	$0,022 \pm 0,002$
RBF SVM	$0,514 \pm 0,005$	$0,063 \pm 0,005$	$0,145 \pm 0,022$	$0,058 \pm 0,004$	$0,027 \pm 0,001$
Adaboost	$0,482 \pm 0,006$	$0,036 \pm 0,0$	$0,038 \pm 0,0$	$0,04 \pm 0,0$	$0,029 \pm 0,0$
Random Forest	$0,876 \pm 0,005$	$0,546 \pm 0,012$	$0,604 \pm 0,021$	$0,529 \pm 0,018$	$0,011 \pm 0,0$

CUADRO 6.117. Resultados de Social con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,318 \pm 0,016$	$0,093 \pm 0,011$	$0,183 \pm 0,013$	$0,079 \pm 0,01$	$0,039 \pm 0,001$
Decision Tree	$0,525 \pm 0,009$	$0,212 \pm 0,015$	$0,239 \pm 0,02$	$0,199 \pm 0,016$	$0,028 \pm 0,001$
Naive Bayes	$0,39 \pm 0,009$	$0,102 \pm 0,012$	$0,175 \pm 0,042$	$0,089 \pm 0,007$	$0,034 \pm 0,001$
Linear SVM	$0,451 \pm 0,007$	$0,179 \pm 0,009$	$0,165 \pm 0,01$	$0,218 \pm 0,017$	$0,04 \pm 0,001$
Neural Net	$0,5 \pm 0,017$	$0,206 \pm 0,012$	$0,232 \pm 0,029$	$0,204 \pm 0,006$	$0,03 \pm 0,002$
RBF SVM	$0,427 \pm 0,006$	$0,034 \pm 0,002$	$0,073 \pm 0,024$	$0,037 \pm 0,001$	$0,032 \pm 0,001$
Adaboost	$0,45 \pm 0,008$	$0,034 \pm 0,001$	$0,037 \pm 0,0$	$0,038 \pm 0,0$	$0,03 \pm 0,001$
Random Forest	$0,577 \pm 0,006$	$0,144 \pm 0,01$	$0,314 \pm 0,045$	$0,116 \pm 0,008$	$0,024 \pm 0,0$

CUADRO 6.118. Resultados de Social con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,323 \pm 0,01$	$0,134 \pm 0,009$	$0,182 \pm 0,027$	$0,159 \pm 0,004$	$0,049 \pm 0,004$
Decision Tree	$0,736 \pm 0,008$	$0,349 \pm 0,029$	$0,372 \pm 0,035$	$0,346 \pm 0,032$	$0,019 \pm 0,001$
Naive Bayes	$0,43 \pm 0,011$	$0,121 \pm 0,015$	$0,218 \pm 0,026$	$0,104 \pm 0,011$	$0,032 \pm 0,001$
Linear SVM	$0,556 \pm 0,006$	$0,231 \pm 0,01$	$0,221 \pm 0,013$	$0,266 \pm 0,007$	$0,033 \pm 0,001$
Neural Net	$0,596 \pm 0,004$	$0,285 \pm 0,01$	$0,316 \pm 0,014$	$0,283 \pm 0,01$	$0,026 \pm 0,0$
RBF SVM	$0,446 \pm 0,008$	$0,04 \pm 0,002$	$0,106 \pm 0,002$	$0,042 \pm 0,001$	$0,031 \pm 0,001$
Adaboost	$0,475 \pm 0,01$	$0,036 \pm 0,001$	$0,041 \pm 0,01$	$0,04 \pm 0,001$	$0,029 \pm 0,001$
Random Forest	$0,69 \pm 0,004$	$0,309 \pm 0,013$	$0,438 \pm 0,027$	$0,266 \pm 0,009$	$0,019 \pm 0,0$

CUADRO 6.119. Resultados de Social con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,231 \pm 0,016$	$0,068 \pm 0,007$	$0,329 \pm 0,033$	$0,054 \pm 0,004$	$0,061 \pm 0,001$
Decision Tree	$0,186 \pm 0,006$	$0,258 \pm 0,01$	$0,256 \pm 0,015$	$0,267 \pm 0,005$	$0,075 \pm 0,001$
Naive Bayes	$0,144 \pm 0,004$	$0,111 \pm 0,005$	$0,095 \pm 0,006$	$0,163 \pm 0,008$	$0,134 \pm 0,002$
Linear SVM	$0,274 \pm 0,005$	$0,301 \pm 0,012$	$0,313 \pm 0,013$	$0,294 \pm 0,012$	$0,063 \pm 0,001$
Neural Net	$0,328 \pm 0,005$	$0,372 \pm 0,012$	$0,477 \pm 0,016$	$0,315 \pm 0,012$	$0,051 \pm 0,0$
RBF SVM	$0,24 \pm 0,009$	$0,084 \pm 0,002$	$0,472 \pm 0,05$	$0,056 \pm 0,001$	$0,052 \pm 0,001$
Adaboost	$0,296 \pm 0,009$	$0,269 \pm 0,011$	$0,489 \pm 0,01$	$0,196 \pm 0,009$	$0,049 \pm 0,001$
Random Forest	$0,275 \pm 0,008$	$0,171 \pm 0,005$	$0,498 \pm 0,032$	$0,114 \pm 0,003$	$0,049 \pm 0,001$

CUADRO 6.120. Resultados de Society con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,164 \pm 0,004$	$0,139 \pm 0,006$	$0,547 \pm 0,018$	$0,086 \pm 0,004$	$0,053 \pm 0,001$
Decision Tree	$0,534 \pm 0,009$	$0,463 \pm 0,012$	$0,457 \pm 0,018$	$0,475 \pm 0,01$	$0,042 \pm 0,002$
Naive Bayes	$0,12 \pm 0,007$	$0,115 \pm 0,004$	$0,099 \pm 0,004$	$0,168 \pm 0,008$	$0,132 \pm 0,003$
Linear SVM	$0,188 \pm 0,01$	$0,285 \pm 0,007$	$0,24 \pm 0,006$	$0,371 \pm 0,011$	$0,081 \pm 0,001$
Neural Net	$0,258 \pm 0,004$	$0,342 \pm 0,009$	$0,314 \pm 0,01$	$0,391 \pm 0,011$	$0,064 \pm 0,001$
RBF SVM	$0,117 \pm 0,007$	$0,041 \pm 0,006$	$0,379 \pm 0,044$	$0,028 \pm 0,003$	$0,054 \pm 0,001$
Adaboost	$0,464 \pm 0,008$	$0,418 \pm 0,014$	$0,622 \pm 0,023$	$0,325 \pm 0,012$	$0,037 \pm 0,001$
Random Forest	$0,507 \pm 0,006$	$0,374 \pm 0,013$	$0,658 \pm 0,05$	$0,283 \pm 0,009$	$0,035 \pm 0,001$

CUADRO 6.121. Resultados de Society con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,17 \pm 0,016$	$0,082 \pm 0,002$	$0,379 \pm 0,033$	$0,064 \pm 0,004$	$0,066 \pm 0,003$
Decision Tree	$0,208 \pm 0,003$	$0,276 \pm 0,012$	$0,279 \pm 0,011$	$0,276 \pm 0,015$	$0,07 \pm 0,001$
Naive Bayes	$0,148 \pm 0,008$	$0,113 \pm 0,002$	$0,094 \pm 0,002$	$0,166 \pm 0,005$	$0,133 \pm 0,004$
Linear SVM	$0,24 \pm 0,01$	$0,274 \pm 0,005$	$0,258 \pm 0,005$	$0,3 \pm 0,005$	$0,072 \pm 0,001$
Neural Net	$0,293 \pm 0,008$	$0,336 \pm 0,007$	$0,367 \pm 0,007$	$0,319 \pm 0,008$	$0,058 \pm 0,001$
RBF SVM	$0,186 \pm 0,004$	$0,055 \pm 0,003$	$0,382 \pm 0,03$	$0,036 \pm 0,002$	$0,054 \pm 0,0$
Adaboost	$0,302 \pm 0,007$	$0,294 \pm 0,011$	$0,51 \pm 0,03$	$0,216 \pm 0,009$	$0,048 \pm 0,0$
Random Forest	$0,289 \pm 0,005$	$0,196 \pm 0,009$	$0,472 \pm 0,022$	$0,134 \pm 0,006$	$0,048 \pm 0,001$

CUADRO 6.122. Resultados de Society con Binary Relevance- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,162 \pm 0,009$	$0,133 \pm 0,007$	$0,538 \pm 0,04$	$0,081 \pm 0,004$	$0,054 \pm 0,001$
Decision Tree	$0,407 \pm 0,014$	$0,396 \pm 0,014$	$0,389 \pm 0,016$	$0,408 \pm 0,012$	$0,052 \pm 0,001$
Naive Bayes	$0,131 \pm 0,004$	$0,113 \pm 0,006$	$0,095 \pm 0,005$	$0,165 \pm 0,01$	$0,132 \pm 0,001$
Linear SVM	$0,172 \pm 0,008$	$0,272 \pm 0,01$	$0,228 \pm 0,008$	$0,358 \pm 0,016$	$0,084 \pm 0,001$
Neural Net	$0,239 \pm 0,009$	$0,329 \pm 0,013$	$0,295 \pm 0,014$	$0,392 \pm 0,007$	$0,068 \pm 0,001$
RBF SVM	$0,113 \pm 0,005$	$0,037 \pm 0,005$	$0,314 \pm 0,051$	$0,025 \pm 0,003$	$0,055 \pm 0,001$
Adaboost	$0,393 \pm 0,014$	$0,384 \pm 0,012$	$0,588 \pm 0,006$	$0,294 \pm 0,011$	$0,041 \pm 0,001$
Random Forest	$0,41 \pm 0,009$	$0,325 \pm 0,012$	$0,612 \pm 0,054$	$0,242 \pm 0,009$	$0,039 \pm 0,001$

CUADRO 6.123. Resultados de Society con Binary Relevance- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,257 \pm 0,015$	$0,078 \pm 0,003$	$0,276 \pm 0,024$	$0,067 \pm 0,002$	$0,067 \pm 0,002$
Decision Tree	$0,279 \pm 0,01$	$0,215 \pm 0,008$	$0,228 \pm 0,01$	$0,207 \pm 0,007$	$0,071 \pm 0,001$
Naive Bayes	$0,271 \pm 0,01$	$0,105 \pm 0,007$	$0,212 \pm 0,035$	$0,086 \pm 0,005$	$0,064 \pm 0,001$
Linear SVM	$0,269 \pm 0,008$	$0,188 \pm 0,007$	$0,162 \pm 0,007$	$0,246 \pm 0,014$	$0,102 \pm 0,004$
Neural Net	$0,287 \pm 0,025$	$0,233 \pm 0,024$	$0,248 \pm 0,027$	$0,232 \pm 0,021$	$0,074 \pm 0,005$
RBF SVM	$0,315 \pm 0,006$	$0,061 \pm 0,001$	$0,375 \pm 0,024$	$0,057 \pm 0,001$	$0,059 \pm 0,001$
Adaboost	$0,305 \pm 0,005$	$0,044 \pm 0,001$	$0,053 \pm 0,004$	$0,05 \pm 0,0$	$0,06 \pm 0,001$
Random Forest	$0,37 \pm 0,007$	$0,186 \pm 0,011$	$0,384 \pm 0,031$	$0,142 \pm 0,007$	$0,054 \pm 0,001$

CUADRO 6.124. Resultados de Society con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,178 \pm 0,008$	$0,124 \pm 0,012$	$0,236 \pm 0,03$	$0,152 \pm 0,008$	$0,103 \pm 0,005$
Decision Tree	$0,774 \pm 0,003$	$0,521 \pm 0,025$	$0,525 \pm 0,031$	$0,522 \pm 0,023$	$0,036 \pm 0,001$
Naive Bayes	$0,292 \pm 0,013$	$0,127 \pm 0,008$	$0,246 \pm 0,027$	$0,101 \pm 0,006$	$0,062 \pm 0,001$
Linear SVM	$0,449 \pm 0,007$	$0,282 \pm 0,006$	$0,262 \pm 0,007$	$0,319 \pm 0,01$	$0,073 \pm 0,002$
Neural Net	$0,468 \pm 0,023$	$0,335 \pm 0,035$	$0,371 \pm 0,068$	$0,322 \pm 0,016$	$0,058 \pm 0,007$
RBF SVM	$0,358 \pm 0,012$	$0,102 \pm 0,006$	$0,526 \pm 0,028$	$0,08 \pm 0,003$	$0,056 \pm 0,001$
Adaboost	$0,282 \pm 0,011$	$0,025 \pm 0,0$	$0,018 \pm 0,0$	$0,037 \pm 0,0$	$0,062 \pm 0,001$
Random Forest	$0,729 \pm 0,007$	$0,505 \pm 0,007$	$0,558 \pm 0,008$	$0,478 \pm 0,012$	$0,038 \pm 0,001$

CUADRO 6.125. Resultados de Society con Label Powerset- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,237 \pm 0,009$	$0,116 \pm 0,007$	$0,266 \pm 0,024$	$0,098 \pm 0,006$	$0,072 \pm 0,001$
Decision Tree	$0,284 \pm 0,009$	$0,215 \pm 0,004$	$0,229 \pm 0,003$	$0,207 \pm 0,005$	$0,07 \pm 0,001$
Naive Bayes	$0,27 \pm 0,006$	$0,109 \pm 0,004$	$0,204 \pm 0,02$	$0,089 \pm 0,003$	$0,064 \pm 0,001$
Linear SVM	$0,245 \pm 0,007$	$0,185 \pm 0,006$	$0,161 \pm 0,008$	$0,234 \pm 0,006$	$0,099 \pm 0,002$
Neural Net	$0,278 \pm 0,012$	$0,215 \pm 0,024$	$0,226 \pm 0,035$	$0,217 \pm 0,023$	$0,076 \pm 0,003$
RBF SVM	$0,294 \pm 0,006$	$0,045 \pm 0,003$	$0,296 \pm 0,043$	$0,048 \pm 0,002$	$0,061 \pm 0,001$
Adaboost	$0,306 \pm 0,007$	$0,044 \pm 0,001$	$0,051 \pm 0,001$	$0,05 \pm 0,001$	$0,06 \pm 0,001$
Random Forest	$0,376 \pm 0,006$	$0,192 \pm 0,012$	$0,387 \pm 0,02$	$0,147 \pm 0,009$	$0,054 \pm 0,0$

CUADRO 6.126. Resultados de Society con Label Powerset- tf-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,231 \pm 0,009$	$0,141 \pm 0,012$	$0,226 \pm 0,012$	$0,16 \pm 0,008$	$0,094 \pm 0,004$
Decision Tree	$0,589 \pm 0,013$	$0,349 \pm 0,014$	$0,363 \pm 0,016$	$0,342 \pm 0,016$	$0,053 \pm 0,002$
Naive Bayes	$0,288 \pm 0,006$	$0,13 \pm 0,008$	$0,244 \pm 0,036$	$0,104 \pm 0,004$	$0,063 \pm 0,001$
Linear SVM	$0,381 \pm 0,01$	$0,246 \pm 0,008$	$0,228 \pm 0,008$	$0,284 \pm 0,01$	$0,08 \pm 0,001$
Neural Net	$0,419 \pm 0,018$	$0,298 \pm 0,027$	$0,324 \pm 0,045$	$0,291 \pm 0,018$	$0,062 \pm 0,004$
RBF SVM	$0,311 \pm 0,009$	$0,057 \pm 0,001$	$0,433 \pm 0,017$	$0,054 \pm 0,001$	$0,06 \pm 0,001$
Adaboost	$0,282 \pm 0,012$	$0,025 \pm 0,0$	$0,018 \pm 0,0$	$0,037 \pm 0,0$	$0,062 \pm 0,001$
Random Forest	$0,496 \pm 0,006$	$0,323 \pm 0,008$	$0,471 \pm 0,005$	$0,264 \pm 0,01$	$0,05 \pm 0,001$

CUADRO 6.127. Resultados de Society con Label Powerset- tf-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,017 \pm 0,009$	$0,014 \pm 0,002$	$0,068 \pm 0,027$	$0,024 \pm 0,017$	$0,111 \pm 0,014$
Decision Tree	$0,167 \pm 0,003$	$0,492 \pm 0,004$	$0,52 \pm 0,009$	$0,469 \pm 0,003$	$0,082 \pm 0,0$
Linear SVM	$0,22 \pm 0,004$	$0,492 \pm 0,004$	$0,542 \pm 0,006$	$0,456 \pm 0,005$	$0,073 \pm 0,001$
Neural Net	$0,265 \pm 0,005$	$0,527 \pm 0,004$	$0,653 \pm 0,011$	$0,458 \pm 0,011$	$0,064 \pm 0,0$
RBF SVM	$0,159 \pm 0,002$	$0,144 \pm 0,005$	$0,59 \pm 0,01$	$0,113 \pm 0,003$	$0,076 \pm 0,0$
Adaboost	$0,24 \pm 0,006$	$0,533 \pm 0,007$	$0,653 \pm 0,01$	$0,46 \pm 0,007$	$0,066 \pm 0,0$
Random Forest	$0,162 \pm 0,003$	$0,304 \pm 0,004$	$0,648 \pm 0,006$	$0,215 \pm 0,004$	$0,077 \pm 0,001$

CUADRO 6.128. Resultados de tmc2007 con Binary Relevance- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,021 \pm 0,01$	$0,013 \pm 0,001$	$0,183 \pm 0,023$	$0,034 \pm 0,014$	$0,12 \pm 0,01$
Decision Tree	$0,68 \pm 0,003$	$0,773 \pm 0,002$	$0,792 \pm 0,001$	$0,755 \pm 0,005$	$0,026 \pm 0,001$
Linear SVM	$0,216 \pm 0,004$	$0,527 \pm 0,004$	$0,484 \pm 0,007$	$0,581 \pm 0,006$	$0,076 \pm 0,0$
Neural Net	$0,282 \pm 0,004$	$0,544 \pm 0,006$	$0,547 \pm 0,01$	$0,55 \pm 0,009$	$0,063 \pm 0,0$
RBF SVM	$0,05 \pm 0,002$	$0,089 \pm 0,003$	$0,36 \pm 0,044$	$0,065 \pm 0,002$	$0,078 \pm 0,001$
Adaboost	$0,6 \pm 0,007$	$0,717 \pm 0,006$	$0,817 \pm 0,003$	$0,647 \pm 0,008$	$0,033 \pm 0,001$
Random Forest	$0,63 \pm 0,007$	$0,595 \pm 0,006$	$0,943 \pm 0,006$	$0,482 \pm 0,006$	$0,033 \pm 0,001$

CUADRO 6.129. Resultados de tmc2007 con Binary Relevance- bin-rfl

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,023 \pm 0,005$	$0,021 \pm 0,009$	$0,104 \pm 0,074$	$0,057 \pm 0,018$	$0,137 \pm 0,011$
Decision Tree	$0,175 \pm 0,002$	$0,39 \pm 0,004$	$0,395 \pm 0,006$	$0,386 \pm 0,004$	$0,096 \pm 0,001$
Linear SVM	$0,126 \pm 0,003$	$0,278 \pm 0,006$	$0,299 \pm 0,014$	$0,322 \pm 0,006$	$0,156 \pm 0,007$
Neural Net	$0,175 \pm 0,021$	$0,37 \pm 0,018$	$0,364 \pm 0,022$	$0,387 \pm 0,012$	$0,1 \pm 0,006$
RBF SVM	$0,166 \pm 0,005$	$0,116 \pm 0,005$	$0,371 \pm 0,03$	$0,103 \pm 0,003$	$0,085 \pm 0,0$
Adaboost	$0,106 \pm 0,003$	$0,05 \pm 0,001$	$0,069 \pm 0,018$	$0,054 \pm 0,0$	$0,088 \pm 0,001$
Random Forest	$0,205 \pm 0,007$	$0,365 \pm 0,007$	$0,471 \pm 0,005$	$0,32 \pm 0,008$	$0,087 \pm 0,001$

CUADRO 6.130. Resultados de tmc2007 con Label Powerset- bin-idf

Classifier	Accuracy	F_1 -score	precision	recall	hamming loss
Knn	$0,028 \pm 0,005$	$0,023 \pm 0,011$	$0,112 \pm 0,062$	$0,058 \pm 0,019$	$0,135 \pm 0,011$
Decision Tree	$0,934 \pm 0,002$	$0,849 \pm 0,005$	$0,851 \pm 0,005$	$0,849 \pm 0,007$	$0,016 \pm 0,001$
Linear SVM	$0,243 \pm 0,007$	$0,336 \pm 0,006$	$0,347 \pm 0,014$	$0,353 \pm 0,005$	$0,124 \pm 0,004$
Neural Net	$0,323 \pm 0,021$	$0,453 \pm 0,044$	$0,501 \pm 0,059$	$0,427 \pm 0,034$	$0,077 \pm 0,007$
RBF SVM	$0,233 \pm 0,004$	$0,185 \pm 0,006$	$0,466 \pm 0,02$	$0,155 \pm 0,004$	$0,076 \pm 0,0$
Adaboost	$0,156 \pm 0,003$	$0,054 \pm 0,001$	$0,083 \pm 0,018$	$0,058 \pm 0,0$	$0,085 \pm 0,0$
Random Forest	$0,891 \pm 0,004$	$0,805 \pm 0,007$	$0,817 \pm 0,006$	$0,799 \pm 0,008$	$0,022 \pm 0,001$

CUADRO 6.131. Resultados de tmc2007 con Label Powerset- bin-rfl

Bibliografía

- [1] R. Alfaro and H. Allende. Text representation in multi-label classification: Two new input representations. in: Dobnikar a., lotric u., ster b. (eds) adaptive and natural computing algorithms. *ICANNGA*, 2011.
- [2] R. Alfaro and H. Allende. Clasificación de textos multi-etiquetados con modelo bernoulli multi-variado y representación dependiente de la etiqueta. *Signos - Estudios de lingüística*, 53 (104):549–567, 2020.
- [3] R. Ñanculef, C. Concha, H. Allende, D. Candell, and C. Moraga. Ad-svms: A light extension of svms for multiclassification. *International Journal of Hybrid Intelligence Systems*, Vol. 6 (2):69–79, 2009.
- [4] Ch. M. Bishop. Pattern recognition and machine learning. *Springer-Verlag Berlin, Heidelberg*, 2006.
- [5] P. McCullagh and J.A. Nelder. Generalized linear models. (*second ed*), *Chapman Hall, New York*, 1989.
- [6] Q. Li, H. Peng, and J. Li. A survey on text classification: From shallow to deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31, 2020.
- [7] M. E. Maron. Automatic indexing: An experimental inquiry. *Journal of the ACM*, 1961.
- [8] T.M. Cover and P.E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13:21–27, 1967.
- [9] Th. Joachims. Text categorization with support vector machines: Learning with many relevant features. *European Conference on Machine Learning*, pages 137–142, 1998.
- [10] Th. Joachims. Learning to classify text using support vector machines: Methods, theory, and algorithms. *Dordrecht: Kluwer Academic*, 2002.
- [11] G. Anthes. Deep learning comes of age. *commun. ACM*, 56:13–15, 2013.
- [12] A. Severyn and A. Moschitti. Learning to rank short text pairs with convolutional deep neural networks. *In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile*, page 373–382, 2015.
- [13] K. Samir and Y. Takehisa. A review on the application of deep learning in system health management. *Mechanical Systems and Signal Processing*, 107:241–265, 2018.
- [14] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, 2019.

- [15] J. Zeng, B. Ustun, and C. Rudin. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 180(3):689–722, Sep 2016.
- [16] K. Kowsari, K. Meimandi, M. Heidarysafa, S. Mendu, L.E. Barnes, and D.E. Brown. Text classification algorithms: A survey. *Machine Learning on Scientific Data and Information*, 2019.
- [17] C. Valle. Ensemble learning with locally coupled learners. *Dissertation, Universidad Técnica Federico Santa María*, 2014.
- [18] R. G. Benjamín. Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty. 24 (1):63–88, 2012.
- [19] J. Lavalle, M. Montes, L. Villaseñor, H. Jiménez, and E. Bárcenas. Supervised and traditional term weighting methods for automatic text categorization. *Studia Logica*, 106 (2):371–395, 2018.
- [20] R. M. Ortega, A. P. López-Monroy, A. Franco, and M. Montes-y Gómez. Emphasizing personal information for author profiling: New approaches for term selection and weighting. *Knowledge-Based Systems*, pages 169–181, 2018.
- [21] A. Chatterjee, U. Gupta, M. K. Chinnakotla, R. Srikanth, M. Galley, and P. Agrawal. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior*, pages 309–317, 2019.
- [22] E. Fink. Automatic evaluation and selection of problem-solving methods: Theory and experiments. *Journal of Experimental and Theoretical Artificial Intelligence*, 16 (2):73–105, 2004.
- [23] A. I. Kadhim. Term weighting for feature extraction on twitter: A comparison between bm25 and tf-idf. *International Conference on Advanced Science and Engineering (ICOASE)*, pages 124–128, 2019.
- [24] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Survey*, 34 (1):1–473, 2002.
- [25] M. Keikha, N. Razavian, F. Oroumchian, and H. S. Razi. Document representation and quality of text: An analysis. en survey of text mining ii: Clustering, classification, and retrieval. pages 135–168, 2008.
- [26] C. Manning and H. Schütze. Foundations of statistical natural language processing. *The MIT Press.*, 1999.
- [27] A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. *AAAI-98 workshop on learning for text categorization, Madison*, pages 41–48, 1998.
- [28] E. Leopold and J. Kindermann. Text categorization with support vector machines. how to represent texts in input space? *Machine Learning*, 46 (1-3):423, 444, 2002.
- [29] M. Lan, C. L. Tan, J. Su, and Y. Lu. Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:721–735, 2009.
- [30] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management: An International Journal*, pages 13–523, 1988.

- [31] K. Kowsari, M. K. Jafari, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown. Text classification algorithms: A survey. *10* (4):150, 2019.
- [32] G. Tsoumakas and I. Katakis. Multi label classification: An overview. *A International Journal of Data Warehouse and Mining*, 3 (3):1–13, 2007.
- [33] S. Lee and J. Jiang. Multilabel text categorization based on fuzzy relevance clustering. *Fuzzy Systems IEEE Transactions*, 22 (6):1457, 1471, 2014.
- [34] J. Nam, J. Kim, E. Mencía, I. Gurevych, and J. Fürnkranz. Large-scale multi-label text classification -revisiting neural networks. *Machine Learning and Knowledge Discovery in Databases*, pages 437–452, 2014.
- [35] E. Giunchiglia and Th. Lukasiewicz. Multi-label classification neural networks with hard logical constraints. *2103.13427v1*, 2021.
- [36] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arxiv preprint arxiv:1810.04805. 2018.
- [37] A. Pal, M. Selvakumar, and M. Sankarasubbu. Magnet: Multi-label text classification using attention-based graph neural network. arxiv preprint arxiv:2003.11644. 2020.
- [38] Y. Murawaki. Global model for hierarchical multi-label text classification. *International Joint Conference on Natural Language Processing*, pages 46–54, 2013.
- [39] J. Liu, W-Ch. Chang, Y. Wu, and Y. Yang. Deep learning for extreme multi-label text classification. *40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*, pages 115–124, 2017.
- [40] Z. Yang, Y. Han, G. Yu, Q. Yang, and X. Zhang. Prototypical networks for multi-label learning. *arXiv:1911.07203*, 2020.
- [41] J. Read, B. Pfahringer, G. Holmes, and Frank E. Classifier chains for multi label classification. *Machine Learning*, pages 333–359, 2011.
- [42] L. I. Kuncheva. Combining pattern classifiers: methods and algorithms. *Second edition, New Jersey*, 2014.
- [43] L. Rokach. Taxonomy for characterizing ensemble methods in classification tasks: a review and annotated bibliography. *Computational Statistics and Data Analysis*, 53:4046–4072, 2009.