Detección de Spam en Mensajes con Random Forest y Regresión Logística

Christian Servando Garza Gonzalez
Maestría en Ciencia de Datos
christian.garzagn@uanl.edu.mx

***Resumen*—En este trabajo se presenta un modelo de clasificación para la detección de mensajes de texto (SMS) spam. Se comparan dos enfoques: un modelo basado en *Random Forest* y otro en *Regresión Logística*, evaluando distintos umbrales de decisión y la influencia de la ponderación de clases (`class_weight`). Los resultados muestran la efectividad de ambos métodos y cómo ajustar parámetros para mejorar indicadores de desempeño como precisión, exhaustividad (recall) y AUC.**

I. Introducción

La proliferación de mensajes de texto no deseados (spam) representa un problema de gran relevancia en la actualidad. Distintos métodos se han propuesto para mitigar su impacto, entre ellos, algoritmos de clasificación basados en características textuales. En este trabajo, se presenta la creación de dos modelos de clasificación (Random Forest y Regresión Logística) y se analiza la influencia de ajustar el parámetro `class_weight` para contrarrestar el posible desbalance de clases (spam vs. ham).

II. Metodología

II-A. Descripción del dataset

Para este estudio, se empleó el conjunto de datos *SMS Spam Collection Dataset*. Tras la limpieza inicial, se eliminaron columnas irrelevantes y se generaron nuevas características, entre ellas:

- `word_count`: Número de palabras en el mensaje.
- `char_count`: Cantidad total de caracteres.
- `uppercase_count`: Número de letras en mayúsculas.
- `special_char_count`: Número de caracteres especiales.
- `number_count`: Cantidad de dígitos.
- `spam_keyword_count`: Frecuencia de palabras típicas de spam.
- `has_spam_keyword`: Indicador binario (0/1) de presencia de keywords spam.

La variable objetivo es `label`, codificada como 0 (*ham*) y 1 (*spam*). El conjunto se dividió en datos de entrenamiento (80 %) y prueba (20 %) de forma estratificada, y se aplicó *StandardScaler* a las características numéricas.

II-B. Modelos Utilizados

II-B1. Random Forest: El modelo de *Random Forest* consiste en un conjunto de árboles de decisión independientes cuyos resultados se combinan para emitir una predicción final. Esta forma de "votación" tiende a mejorar la estabilidad y exactitud respecto a usar un único árbol. Para nuestro caso, se evaluó el desempeño del modelo bajo distintos umbrales de decisión (0.2, 0.5 y 0.8), lo que permitió observar el compromiso entre detectar la mayor cantidad de spam (recall) y minimizar falsos positivos (precision). Además, se comparó el escenario con mayor atención a la clase minoritaria (balanceo de clases) frente a la situación estándar (sin balanceo).

II-B2. Regresión Logística: La *Regresión Logística* es un modelo lineal ampliamente utilizado en clasificación, pues ofrece interpretabilidad y suele ser eficiente para identificar la probabilidad de que un mensaje pertenezca a la clase spam. Al igual que con *Random Forest*, exploramos varios umbrales (0.2, 0.5 y 0.8) y valoramos el efecto de asignar un peso adicional a la clase minoritaria (balanceo de clases). Para cada configuración, se calculó también el área bajo la curva ROC (AUC), que sirve para medir la capacidad del modelo de distinguir entre mensajes legítimos y spam en un rango amplio de umbrales.

III. Resultados

III-A. Random Forest

En la Tabla I se observa la matriz de confusión desglosada (TN, FP, FN, TP), así como la *precision* y *recall* promedio en porcentaje, para cada combinación de balanceo de clases y umbral.

Cuadro I
Resultados con Random Forest

class_weight	threshold	TN	FP	FN	TP	Prec (%)	Recall (%)
balanced	0.2	953	13	13	136	91.28	91.28
balanced	0.5	964	2	19	130	98.48	87.25
balanced	0.8	966	0	37	112	100.00	75.17
None	0.2	950	16	12	137	89.54	91.95
None	0.5	964	2	18	131	98.50	87.92
None	0.8	966	0	32	117	100.00	78.52

Al aumentar el umbral de 0.2 a 0.8, el modelo se vuelve más "estricto" a la hora de clasificar un mensaje como spam. Esto se traduce en menos falsos positivos (FP) (es decir, menos correos legítimos mal etiquetados), lo que a su vez mejora la precisión. Sin embargo, también aumenta el número de falsos negativos (FN), lo que significa que más correos de spam pasan inadvertidos; en otras palabras, disminuye el recall.

III-B. Regresión Logística

La Tabla II presenta los resultados obtenidos con Regresión Logística para los mismos umbrales y valores de balanceo de clases.

Cuadro II
RESULTADOS CON REGRESIÓN LOGÍSTICA

class_weight	threshold	TN	FP	FN	TP	Prec (%)	Recall (%)
None	0.2	955	11	17	132	92.31	88.59
None	0.5	962	4	24	125	96.90	83.89
None	0.8	965	1	28	121	99.18	81.21
balanced	0.2	895	71	5	144	66.98	96.64
balanced	0.5	950	16	13	136	89.47	91.28
balanced	0.8	959	7	20	129	94.85	86.58

Con el balanceo de clases activado y un umbral bajo (0.2), el modelo pone mayor énfasis en detectar la clase spam, alcanzando un recall de (96,64 %). Sin embargo, esto provoca un número significativamente más alto de falsos positivos (FP=71), lo que reduce la precisión (66,98 %).

Al elevar el umbral a (0.5), se logra un equilibrio más favorable entre ambas métricas: recall de (91,28 %) y una precisión cercana al (89,47 %). Esto significa que, si bien disminuye la detección de spam, también se reduce el riesgo de etiquetar mensajes legítimos como spam.

Por su parte, cuando se desactiva el balanceo de clases, el modelo favorece la clase mayoritaria (ham) y reduce la aparición de falsos positivos (por ejemplo, con umbral de 0.8 solo hay 1 FP), pero disminuye levemente su capacidad de capturar spam (recall baja a (81,21 %)).

Así, cada combinación de umbral y balance de clases ofrece un balance diferente entre evitar clasificaciones erróneas de correos legítimos (FP) y recuperar la mayor parte de los correos spam (FN). La elección de una configuración depende de la prioridad en la aplicación: maximizar recall (ser muy sensibles al spam) o reforzar la precisión para minimizar falsos positivos.

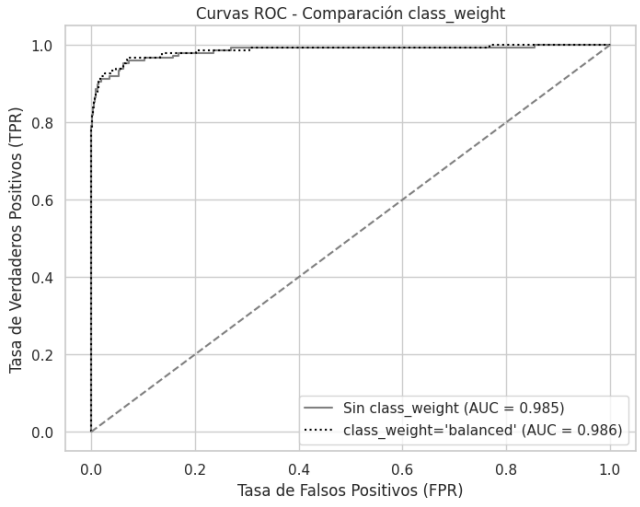

Figura 1. Comparación de curvas ROC para Regresión Logística (Sin balanceo vs. Balanceado).

III-C. Curvas ROC

La Figura 1 muestra la comparación de las curvas ROC para Regresión Logística con y sin balanceo. Ambas soluciones presentan un rendimiento alto, con AUC por arriba de 0.98. La pequeña diferencia radica en que el balance de clases tiende a mejorar la detección de la clase minoritaria.

III-D. Importancia de las Variables

Para analizar qué variables influyen más en la clasificación de mensajes como spam o ham, se comparó la importancia relativa de cada característica en ambos modelos. En el caso de *Random Forest*, la importancia se mide en función de su contribución a la reducción de impureza en los árboles de decisión. En *Regresión Logística*, se calcularon los coeficientes absolutos normalizados para hacerlos comparables. La Tabla III muestra los resultados.

Cuadro III
IMPORTANCIA DE LAS VARIABLES EN RANDOM FOREST Y REGRESIÓN LOGÍSTICA

Variable	Random Forest (%)	Regresión Logística (%)
number_count	49.19	30.68
uppercase_count	18.18	0.54
spam_keyword_count	9.41	37.31
char_count	9.11	3.23
word_count	6.28	14.45
has_spam_keyword	3.98	4.70
special_char_count	3.84	9.08

La Figura 2 muestra una comparación visual de la importancia de cada variable en ambos modelos.

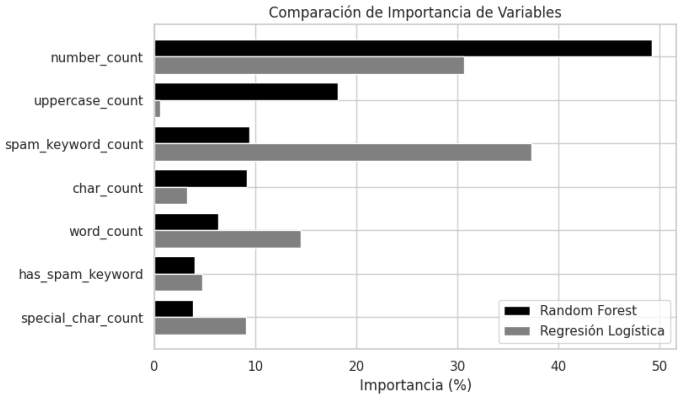

Figura 2. Comparación de importancia de variables entre Random Forest y Regresión Logística.

Como se observa, el número de dígitos en un mensaje es la característica más relevante en ambos modelos, aunque tiene un peso mayor en Random Forest (49,19 %) en comparación con Regresión Logística (30,68 %). Sin embargo, la presencia de palabras clave de spam tiene mayor influencia en la Regresión Logística (37,31 %) que en Random Forest (9,41 %). Esto sugiere que cada modelo prioriza distintos patrones en la detección de spam. Además, la cantidad de caracteres en mayúsculas es más relevante en Random Forest que en Regresión Logística.

Estos resultados refuerzan la idea de que modelos basados en árboles de decisión y modelos lineales pueden enfocarse en diferentes señales dentro de los datos, lo que podría motivar el uso de enfoques híbridos para mejorar la detección de spam.

IV. Discusión

Los resultados muestran que ambos modelos (Random Forest y Regresión Logística) ofrecen métricas de desempeño muy altas (precisión y recall superiores al 80 % - 90 %, AUC cercano a 0.98). La inclusión de balanceo de clases facilita aumentar el recall para la clase spam (positiva), aunque puede elevar la tasa de falsos positivos.

En cuanto al umbral de decisión, valores de 0.5 suelen dar un equilibrio razonable entre precisión y recall. Umbrales bajos (0.2) maximizan la captura de spam a costa de etiquetar más mensajes legítimos (ham) como spam. Umbrales altos (0.8) reducen casi a cero los falsos positivos, pero sacrifican la detección de spam (disminuye el recall).

V. Conclusiones

Mediante el análisis de diferentes escenarios (uso o no de balance de clases y distintos umbrales de clasificación), se observa que:

- Tanto Random Forest como Regresión Logística consiguen un rendimiento elevado, con **AUC** que supera el 0.98.
- Ajustar el **umbral** es clave para controlar el balance entre precisión y recall.
- La opción de considerar un balanceo de clases incrementa la capacidad de recuperar spam (mayor recall), pero puede derivar en más falsos positivos.

Este enfoque puede servir de base para sistemas de detección de spam en SMS, siendo posible extenderlo a otros tipos de mensajes (correo electrónico, chats en aplicaciones de mensajería, redes sociales, etc.).

Finalmente, desde un punto de vista práctico, umbrales más bajos resultan útiles cuando la prioridad es capturar la mayor cantidad de spam posible (aun a costa de algunos falsos positivos), mientras que umbrales más altos minimizan la clasificación errónea de correos legítimos, pero pueden pasar por alto mayor spam. De esta manera, cada escenario refleja un compromiso entre precisión y recall, y la elección depende de las necesidades específicas del sistema de filtrado (por ejemplo, la tolerancia a recibir spam vs. el riesgo de perder información importante marcada erróneamente).

Referencias

[1] P.-A. Chirita, J. Diederich, and W. Nejdl, "MailRank: Using ranking for spam detection," *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, 2005.

[2] T. Almeida, J. G. Hidalgo, and A. Yamakami, "Contributions to the study of SMS spam filtering: new collection and results," in *Proceedings of the 11th ACM Symposium on Document Engineering*, 2011.

[3] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, 2011.