Predecir la popularidad de noticias en línea utilizando modelos de clasificación y/o regresión

Calderon, S., Arenas, K., Torres, J.

1. Resumen

2. Introducción

El estudio de la popularidad de las noticias en línea ha cobrado gran relevancia en la última década, ya que los medios compiten por la atención en un entorno digital saturado. La capacidad de predecir qué noticias serán populares es esencial para optimizar estrategias de contenido y aumentar la participación de la audiencia. Para abordar este desafío, se han desarrollado modelos basados en datos históricos y aprendizaje automático que consideran una variedad de características, como el contenido de la noticia, el comportamiento del usuario y las dinámicas de las plataformas de distribución. Los usuarios, quienes son los consumidores de estos medios usualmente comparten, se suscriben e interactúan con la información. La noción de popularidad suele expresarse investigando el número de interacciones en la web y las redes sociales, por ejemplo, la tasa de clics, el número de compartidos, me gusta y retweets [1]. El número de veces que se comparte una noticia es un buen indicador de su popularidad potencial. La predicción de la popularidad de un artículo tiene múltiples aplicaciones en el mundo real, desde el punto de vista de [2], hay dos caminos de predicción más conocidas: el primero relacionado con las características que se conocen luego de la publicación y los que no utilizan dichas características. El primero es el más conocido y más común [3, 4, 5, 6, 7]. De acuerdo a [8], la popularidad de un artículo candidato puede ser estimado utilizando un módulo de predicción y luego un módulo de optimización, según el autor sugiere cambios en el contenido y la estructura del artículo, con el fin de maximizar su popularidad esperada. Existen varios estudios que sugieren, la predicción puede ser alta utilizando los metadatos posteriores a la publicación, dándonos ventaja en la atención recibidas de la información luego de ser recibida [5, 6]. Por el contrario [8], menciona que al utilizar las características de metadatos antes de la publicación de los contenidos resulta ser difícil, aunque la precisión de la predicción esperada es comparativamente baja en el método de antes de la publicación, ya que estamos utilizando sólo características de metadatos en lugar del contenido original de la noticia.

El objetivo de este artículo es desarrollar un modelo de aprendizaje automático que logre predecir la popularidad de los artículos de noticias en línea antes de su publicación. La sección de estado del arte presenta trabajos previos en la materia, incluyendo la metodología y resultados. La sección Diseño del experimento se centra en explicar la metodología de este artículo, incluyendo la descripción del conjunto de datos y los modelos a ser entrenados.

3. Estado del arte

Esta sección proporciona un resumen cohesivo de estudios de investigación enfocados en medir y predecir la popularidad de los artículos de noticias en línea. Los dos primeros estudios abordan los desafíos de predecir la popularidad de las noticias en el momento de su publicación, mientras que los últimos tres profundizan en la aplicación de técnicas de aprendizaje automático en un único conjunto de datos para mejorar las capacidades predictivas de la popularidad de los artículos de noticias.

[9] investiga las dificultades asociadas con predecir la popularidad de los artículos de noticias inmediatamente después de su publicación. Utilizando un conjunto de datos de 13,319 artículos de Yahoo News, los autores critican investigaciones previas, que afirmaban una alta precisión en la predicción de

*Corresponding author

la popularidad de las noticias utilizando características basadas en el contenido. Se argumenta que la tarea es más compleja de lo que se pensaba, principalmente debido a la alta asimetría en la distribución de popularidad. Sus experimentos revelan que los métodos de clasificación y regresión existentes están sesgados hacia la predicción de artículos impopulares, fallando en identificar con precisión los populares, que son cruciales para la identificación temprana. El estudio concluye que los métodos actuales son inadecuados para predecir la popularidad de las noticias desde el inicio utilizando solo características basadas en el contenido, destacando la necesidad de enfoques más robustos.

En [10], los autores abordan el desafío de predecir qué artículos de noticias aparecerán en una lista de "más leídos". Proponen que la popularidad es un concepto relativo influenciado por el atractivo de los artículos publicados simultáneamente. Empleando una función lineal en una representación de bolsa de palabras, utilizan Máquinas de Soporte Vectorial de Ranking (Ranking SVMs) entrenadas en pares de artículos de la misma fecha y medio. El modelo, que utiliza información mínima como contenido textual, fechas de publicación y estado de popularidad, predice con éxito los artículos populares e identifica palabras clave influyentes. Utilizando un conjunto de datos de diez medios importantes en inglés durante un año, el estudio logra precisiones que a menudo superan el 60%, superando los métodos de clasificación binaria. Esta investigación destaca la naturaleza dinámica del atractivo de las noticias y proporciona un marco robusto para predecir la popularidad de las noticias.

En [8], se introduce un Sistema Proactivo de Soporte de Decisiones Inteligente (IDSS) está diseñado para predecir la popularidad de los artículos de noticias en línea antes de su publicación. Utilizando un conjunto de datos de 39,000 artículos del sitio web Mashable, el IDSS aprovecha diversas características, incluyendo contenido digital, popularidad de noticias referenciadas, participación de palabras clave y análisis de sentimientos. El sistema emplea modelos de aprendizaje automático como Bosques Aleatorios (Random Forest), Adaptive Boosting y Máquinas de Soporte Vectorial (SVM) para una tarea de clasificación binaria. El modelo Random Forest alcanza un poder de discriminación del 73%, mientras que un módulo de optimización que utiliza búsqueda local por escalada estocástica mejora la probabilidad de popularidad estimada en 15 puntos porcentuales. El IDSS no solo predice, sino que también sugiere modificaciones en el contenido y la estructura del artículo para mejorar la popularidad esperada, demostrando ser una herramienta valiosa para los autores de noticias en línea.

Usando el mismo conjunto de datos, [11] evalúa diversas técnicas de aprendizaje automático para predecir la popularidad de los artículos de noticias en línea. Se obtuvieron conocimientos iniciales utilizando regresión lineal y logística, logrando la regresión logística una precisión del 66% al categorizar la variable objetivo en categorías binarias. Usando SVM, los autores enfrentaron inicialmente problemas de alto sesgo, pero mostraron mejoras marginales con núcleos más complejos, alcanzando una precisión del 55%. Sin embargo, el modelo de Random Forest emergió como el más efectivo, logrando una precisión del 70% con parámetros óptimos. Al aprovechar múltiples árboles de decisión y subconjuntos de características, Random Forest mitigó eficazmente la varianza, proporcionando las predicciones más precisas.

Finalmente, [12] explora la predicción de la popularidad de artículos de noticias utilizando el conjunto de datos de Mashable mediante diversas técnicas de aprendizaje automático. Se aplicaron métodos de selección de características como la selección univariada, la eliminación recursiva de características y el análisis de componentes principales para identificar las características más relevantes que influyen en la popularidad de los artículos. Evaluaron once modelos de clasificación, incluidos Naïve Bayes, regresión logística, árboles de decisión, redes neuronales, bosques aleatorios y máquinas de vectores de soporte. Entre estos, el método de potenciación del gradiente (gradient boosting) surgió como el modelo más eficaz, logrando una precisión del 79.7%. El estudio concluyó que los métodos en ensamble, en particular gradient boosting, son los que mejor funcionan para predecir la popularidad de los artículos de noticias.

Esta colección de investigaciones destaca las complejidades y avances en la predicción de la popularidad de las noticias en línea. La predicción temprana desde el inicio sigue siendo un desafío debido a las distribuciones de popularidad sesgadas y los sesgos de los modelos. Sin embargo, las técnicas sofisticadas de aprendizaje automático y las estrategias exhaustivas de preprocesamiento muestran prome-

sas para mejorar la precisión de la predicción. La investigación y desarrollo continuos de metodologías robustas son cruciales para futuros avances en este dominio.

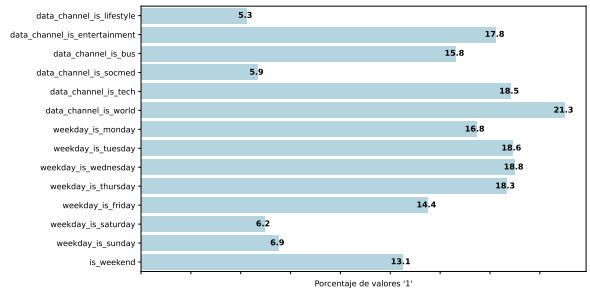
4. Diseño del experimento

Para este estudio, se utilizará el conjunto de datos elaborado en [8]. Cuenta con 59 características y una columna target. esta última contiene información de las veces que el artículo fue compartido (numérica), siendo esta la medición de popularidad. En general, el conjunto de datos permite un análisis exhaustivo de las características del contenido, la participación del usuario y la efectividad de varios tipos de contenido a lo largo de diferentes canales y en el tiempo. A continuación, se presenta una descripción más detallada.

4.1. Descripción del conjunto de datos

El trabajo utiliza los datos del repositorio de aprendizaje automático UCI (https://archive.ics.uci.edu/dataset/332/online+news+popularity). La información fue utilizada para entrenar modelos predictivos de compartidos de noticias de Mashable (www.mashable.com), abarca un período de dos años (7 de enero de 2013 al 7 de enero de 2015) y contiene 58 características heterogéneas. Estas características incluyen la longitud del título y del contenido, el número de imágenes y videos, variables temporales como el día de la semana y la hora de publicación, y metadatos como la popularidad en otras plataformas y el tema de la noticia. También se consideran características sociales, como el número de comentarios y compartidos en redes sociales, y técnicas, como la presencia de multimedia y palabras clave. Este conjunto de datos integral permite un análisis exhaustivo para desarrollar modelos predictivos efectivos que estimen la popularidad de una noticia basada en su probabilidad de ser compartida.

De acuerdo a la página web UCI, el conjunto de datos de Mashable consta de 39 797 instancias con 58 características, que abarcan tipos de datos enteros y reales. Los datos se pueden adaptar para aplicar modelos de clasificación y regresión. Len resumen, los datos proporcionan información sobre diversos aspectos de los artículos de noticias, lo que los hace valiosos para analizar tendencias y predecir métricas de rendimiento en el panorama de los medios digitales.



El conjunto de datos también presenta componentes de análisis semántico latente y puntuaciones de sentimiento, capturando el tono emocional y la objetividad del contenido. Las métricas de participación, como el número de veces que se comparte y las referencias a sí mismo, son cruciales para comprender el alcance e impacto del contenido. Todas estas son variables continuas.

4.2. Metodología

El presente trabajo prioriza la etapa de procesamiento del dataset, paso fundamental antes de realizar el entrenamiento de los datos. El conjunto de datos que actualmente manejamos cuenta con 58 características y 39 mil registros, por lo que claramente se observa la necesidad de reducir el número de caracteristicas. Para un adecuado procesamiento y entrenamiento de datos, el presente trabjo considera los sigientes pasos: 1) Exploración y análisis de datos: en este paso realizaremos la reducción de la dimensionalidad utilizando la herramienta de análisis de principales componentes (PCA) y análisis de correlación de datos, ambos métodos nos ayudaran a visualizar y entender las relaciones y patrones de las variables. 2) Selección y entrenamiento del modelo: En este paso utilizaremos varios modelos de clasificación, incluyendo RandomForestClassifier [13], AdaBoostClassifier [14], LogisticRegression [15] y K-Nearest Neighbors [**?**]. Para optimizar los hiperparámetros y mejorar el rendimiento de estos modelos se empleó GridSearchCV [16], el cual realiza una búsqueda exhaustiva a través de un rango especificado de valores de parámetros. Además, se consideró el modelo GradientBoostingClassifier [17], un método de boosting que, al igual que AdaBoost, combina múltiples modelos débiles para formar un modelo fuerte. Estos enfoques maximizan la precisión y robustez de las predicciones al ajustar finamente los parámetros y combinar varios clasificadores en un esquema de ensamble. 3) Validación y evaluación del modelo: en otras palabras, medir el rendimiento del modelo utilizando datos de

prueba y métricas de evaluación. Para ello, se evaluará el rendimiento de los modelos mediante área bajo la curva (AUC), una métrica que proporciona una medida agregada del rendimiento en todas las posibles clasificaciones. 4) Finalmente se describirá los resultados, interpretación y discusiones: en esta parte nos apoyaremos mediante figuras y gráficos estadísticos y correlacionaremos con anteriores resultados.

4.2.1. Exploración y análisis de datos

En la fase de exploración y análisis de datos, se comienza con el ordenamiento de caracteres, que implica la normalización y limpieza de los datos textuales, como la conversión a minúsculas, la eliminación de espacios vacios y caracteres especiales. Luego, se procede a la eliminación de características irrelevantes, identificando y descartando aquellas que no aportan valor predictivo. A continuación, se realiza la unión de variables, combinando múltiples características en una sola para simplificar el análisis. La creación de una matriz de correlación permite identificar relaciones entre variables, ayudando a detectar características redundantes. Finalmente, se aplica la disminución de dimensionalidad, utilizando técnicas como el Análisis de Componentes Principales (PCA), para reducir el número de características manteniendo la mayor cantidad de información posible. Este proceso asegura que los datos sean limpios, relevantes y manejables, facilitando el desarrollo de modelos predictivos efectivos.

4.2.2. Selección y entrenamiento del modelo

4.2.3. Clasificación Random Forest

Los bosques aleatorios son métodos de aprendizaje en conjunto para la regresión que utilizan múltiples árboles de decisión. El algoritmo extrae muestras bootstrap del conjunto de datos original y genera árboles de regresión no podados para cada muestra. En lugar de elegir el mejor atributo de división, se selecciona aleatoriamente de un conjunto de atributos. La predicción final se obtiene promediando las predicciones de todos los árboles individuales. Se utilizó la librería `scikit-learn`, para implementar el modelo de Random Forest (bosque aleatorio) y fijamos el número de estimadores en 50 estimadores con random_state igual a 42.

4.2.4. AdaBoostClassifier (Adaptive Boosting)

Es un algoritmo de aprendizaje automático utilizado principalmente para tareas de clasificación. AdaBoost funciona construyendo un modelo a partir de un conjunto inicial de datos y luego ajusta iterativamente los pesos de los ejemplos mal clasificados, incrementando la influencia de aquellos que resultaron más difíciles de clasificar en cada iteración. Esto permite que los clasificadores posteriores se centren más en los ejemplos difíciles. Finalmente, los modelos individuales se combinan mediante una votación ponderada para mejorar la precisión global. AdaBoost es conocido por su capacidad para mejorar el rendimiento de modelos débiles y su eficacia en diversos dominios, aunque su rendimiento puede verse afectado por la presencia de ruido en los datos y por la elección de los clasificadores base. Este modelo se implementó utilizando la librería Scikit-learn.

4.2.5. Regresión logística

La regresión logística es una técnica de clasificación utilizada para predecir la probabilidad de pertenencia a una clase binaria. Utiliza la función sigmoide para convertir una combinación lineal de variables independientes en una probabilidad, y ajusta el modelo minimizando la función de costo logloss. La implementación del modelo fue realizando la librería scikit-learn.

4.2.6. K-Nearest Neighbors (KNN)

El modelo K-Nearest Neighbors (KNN) es un algoritmo de aprendizaje supervisado utilizado para clasificación y regresión. Clasifica una observación basándose en las clases de sus K vecinos más cercanos, donde K=10 en este caso, utilizando una medida de distancia como la Euclidiana. En clasificación, asigna la clase más común entre los vecinos, y en regresión, toma el promedio de los valores de los vecinos. Es un método simple y no paramétrico que no hace suposiciones sobre la distribución de los datos, y puede implementarse fácilmente en Python usando la Libreria scikit-learn.

4.2.7. GridSearchCV

En este análisis, se examinan diversos modelos de clasificación para predecir la popularidad de noticias en línea utilizando el dataset de Online News Popularity de Mashable. Se implementan cuatro modelos: RandomForest, AdaBoost, LogisticRegression y K-Nearest Neighbors (KNN). Para cada modelo, se definen conjuntos específicos de hiperparámetros y se emplea GridSearchCV para realizar una búsqueda exhaustiva de los mejores parámetros mediante validación cruzada. Una vez entrenados y

optimizados los modelos, se selecciona el mejor estimador y se evalúa su rendimiento en un conjunto de prueba. La precisión obtenida en este conjunto se utiliza para comparar la eficacia de cada modelo. Este proceso automatiza la selección de hiperparámetros y facilita una comparación objetiva entre diferentes algoritmos de clasificación, proporcionando información valiosa sobre su desempeño en el contexto de la popularidad de noticias en línea. En la tabla **??** se lista los mejores hiperparametros por modelo.

Modelo	**Hiperparámetro**	**Valores**
RandomForest	n_estimators	[10, 20, 50, 100, 200, 400]
AdaBoost	n_estimators	[10, 20, 50, 100, 200, 400]
LogisticRegression	penalty	['l1', 'l2']
	C	[0.001, 0.01, 0.1, 1, 10]
	solver	['liblinear', 'saga']
K-Nearest Neighbors (KNN)	n_neighbors	[3, 5, 7, 9, 11, 13]
	weights	['uniform', 'distance']
	algorithm	['auto', 'ball_tree', 'kd_tree', 'brute']

Table 1: Hiperparámetros probados para cada modelo

5. Experimentación y resultados

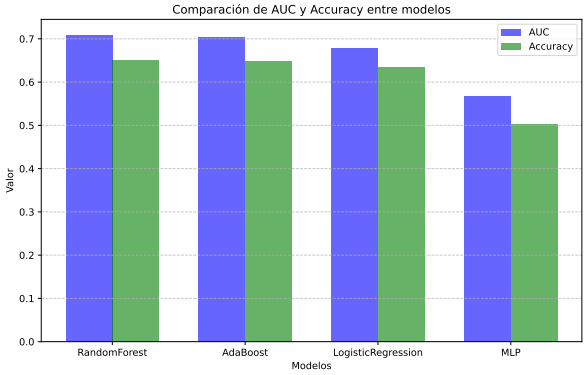
Los resultados muestran que RandomForest y AdaBoost son los modelos con mejor rendimiento general, con AUC (Área bajo la curva ROC) de 0.7095 y 0.7030 respectivamente. RandomForest alcanza una precisión (Accuracy) de 0.6510 y una precisión positiva (Precision) de 0.6447, mientras que AdaBoost logra 0.6488 y 0.6423 respectivamente en estas métricas. LogisticRegression, aunque presenta un AUC de 0.6793, muestra una precisión y precision positiva ligeramente más bajas en comparación con los modelos de ensemble. Por otro lado, MLP (Perceptrón Multicapa) muestra el rendimiento más bajo con un AUC de 0.5679 y una precisión de 0.5034, aunque destaca en recall (Sensibilidad) con 0.7427. Estos resultados resaltan la capacidad superior de los modelos ensemble como RandomForest y AdaBoost para la clasificación, especialmente en términos de AUC y precisión global.

Los resultados de las métricas de evaluación para los diferentes modelos se presentan en la Table 2.

Table 2: Métricas de evaluación de modelos

Model	AUC	Accuracy	Precision	Recall	F
Random Forest	0.7095	0.651	0.6447	0.6406	0.
AdaBoost	0.703	0.6488	0.6423	0.6389	0.
Logistic Regression	0.6793	0.635	0.6286	0.6231	0.
MLP	0.5679	0.5034	0.4954	0.7427	0.

El gráfico de barras muestra la comparación de las métricas de AUC (Área bajo la curva ROC) y Accuracy para cuatro modelos de clasificación: RandomForest, AdaBoost, LogisticRegression y MLP (Perceptrón Multicapa). Cada modelo está representado por barras de diferentes colores: azul para AUC y verde para Accuracy. RandomForest y AdaBoost destacan con altos valores de AUC y Accuracy, seguidos por LogisticRegression con valores ligeramente inferiores. En contraste, MLP muestra un desempeño más bajo en ambas métricas. Esta representación visual facilita la comparación directa del rendimiento relativo de cada modelo en términos de su capacidad predictiva y precisión.



6. Discusión

7. Conclusiones y trabajos futuros

Basado en las métricas evaluadas, los modelos RandomForest y AdaBoost mostraron un rendimiento superior con AUC de 0.7095 y 0.7030 respectivamente, indicando una buena capacidad para clasificar correctamente. LogisticRegression también fue competitivo con un AUC de 0.6793, aunque mostró una precisión ligeramente menor. En contraste, MLP obtuvo un AUC más bajo de 0.5679, pero destacó en sensibilidad (recall) con

0.7427, siendo efectivo en la identificación de casos positivos. En conclusión, RandomForest y AdaBoost son recomendados por su sólido desempeño general en este conjunto de datos, mientras que MLP muestra una fortaleza en la detección de casos positivos.

References

[1] M. T. Uddin, M. J. A. Patwary, T. Ahsan, M. S. Alam, Predicting the popularity of online news from content metadata (2016) 1–5.

[2] A. Tatar, P. Antoniadis, M. D. d. Amorim, S. Fdida, From popularity prediction to ranking online news, Social Network Analysis and Mining 4 (2014) 1–12.

[3] A. Kaltenbrunner, V. Gómez, V. López, Description and prediction of slashdot activity (2007) 57–66.

[4] G. Szabo, B. A. Huberman, Predicting the popularity of online content, Communications of the ACM 53 (8) (2010) 80–88.

[5] J. G. Lee, S. Moon, K. Salamatian, Modeling and predicting the popularity of online contents with cox proportional hazard regression model, Neurocomputing 76 (1) (2012) 134–145.

[6] M. Ahmed, S. Spagna, F. Huici, S. Niccolini, A peek into the future: Predicting the evolution of popularity in user generated content, in: Proceedings of the sixth ACM international conference on Web search and data mining, 2013, pp. 607–616.

[7] A. Tatar, M. D. De Amorim, S. Fdida, P. Antoniadis, A survey on predicting the popularity of web content, Journal of Internet Services and Applications 5 (2014) 1–20.

[8] K. Fernandes, P. Vinagre, P. Cortez, A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News, Springer International Publishing, 2015, pp. 535–546. doi:10.1007/978-3-319-23485-4_53.
URL http://dx.doi.org/10.1007/978-3-319-23485-4_53

[9] I. Arapakis, B. B. Cambazoglu, M. Lalmas, On the Feasibility of Predicting News Popularity at Cold Start, Springer International Publishing, 2014, pp. 290–299. doi:10.1007/978-3-319-13734-6_21.
URL http://dx.doi.org/10.1007/978-3-319-13734-6_21

[10] E. Hensinger, I. Flaounas, N. Cristianini, Modelling and predicting news popularity, Pattern Analysis and Applications 16 (4) (2012) 623–635. doi:10.1007/s10044-012-0314-6.
URL http://dx.doi.org/10.1007/s10044-012-0314-6

[11] H. Ren, Q. Yang, Predicting and evaluating the popularity of online news, 2015.
URL https://api.semanticscholar.org/CorpusID:7149545

[12] A. Khan, G. Worah, M. Kothari, Y. H. Jadhav, A. V. Nimkar, News popularity prediction with ensemble methods of classification, 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (07 2018). doi:10.1109/icccnt.2018.8494095.
URL http://dx.doi.org/10.1109/ICCCNT.2018.8494095

[13] L. Breiman, Random forests, Machine Learning 45 (1) (2001) 5–32.

[14] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, Journal of Computer and System Sciences 55 (1) (1997) 119–139.

[15] D. R. Cox, The regression analysis of binary sequences, Journal of the Royal Statistical Society: Series B (Methodological) 20 (2) (1958) 215–232.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[17] J. H. Friedman, Greedy function approximation: A gradient boosting machine, Annals of Statistics 29 (5) (2001) 1189–1232.