Universidad Nacional Mayor de San Marcos

FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMATICA

Trabajo computacional - Naive Bayes

CURSO: Inteligencia artificial

PROFESOR(A): Rolando Maguiña

INTEGRANTES: Grupo 10

- Adolfo Paucar, Kiltom
- Palomino Julian, Alex Marcelo
- Calderón Zúñiga, Rodrigo Joaquin

Lima, Perú

2025

Índice

Clasificador de Tópicos para Noticias Periodísticas con Naive Bayes

1. Objetivo

El objetivo principal de este trabajo es implementar un clasificador de tópicos para noticias periodísticas utilizando el algoritmo Naive Bayes, específicamente para categorizar documentos del conjunto de datos 20 Newsgroups en 20 temas diferentes.

2. Marco Teórico

El clasificador Naive Bayes es un algoritmo de aprendizaje supervisado basado en el teorema de Bayes, que asume la independencia condicional entre las características dado el valor de la clase. A pesar de esta suposición simplificada, es efectivo para tareas de clasificación de texto[1]. La fórmula para la clasificación es:

$$p(C_k \mid x_1, \ldots, x_n) \propto p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

donde Ck es la clase k, xi son las características (palabras), y p(Ck) es la probabilidad previa de la clase.

En este proyecto, se utiliza el clasificador Multinomial Naive Bayes, ideal para datos discretos como frecuencias de palabras. Para la extracción de características, se emplea TF-IDF (Term Frequency-Inverse Document Frequency), que mide la importancia de una palabra en un documento dentro de un corpus [2]. La fórmula es:

$$\text{tfidf}(t, d, D) = \text{tf}(t, d) \cdot \text{idf}(t, D)$$

donde tf(t, d) es la frecuencia del término t en el documento d, y idf(t,D) = log N/nt ,con N como el número total de documentos y nt como el número de documentos que contienen t.

El conjunto de datos 20 Newsgroups consta de aproximadamente 18,000 posts de noticias divididos en 20 categorías, como “alt.atheism” y “sci.space”. Es un estándar para tareas de clasificación de texto, dividido en conjuntos de entrenamiento (11,314 documentos) y prueba [3].

3. Aplicación

El software desarrollado es un clasificador de tópicos que utiliza Naive Bayes y TFIDF, con una interfaz gráfica de usuario (GUI) implementada en Tkinter. Permite a los usuarios cargar textos, clasificarlos en una de las 20 categorías y visualizar métricas como precisión y matrices de confusión.

En un contexto periodístico, este sistema puede categorizar automáticamente artículos de noticias, facilitando la organización de grandes volúmenes de información. Por ejemplo, un editor podría usarlo para clasificar noticias en categorías como “Política” o “Deportes”, mejorando la eficiencia en la gestión de contenido.

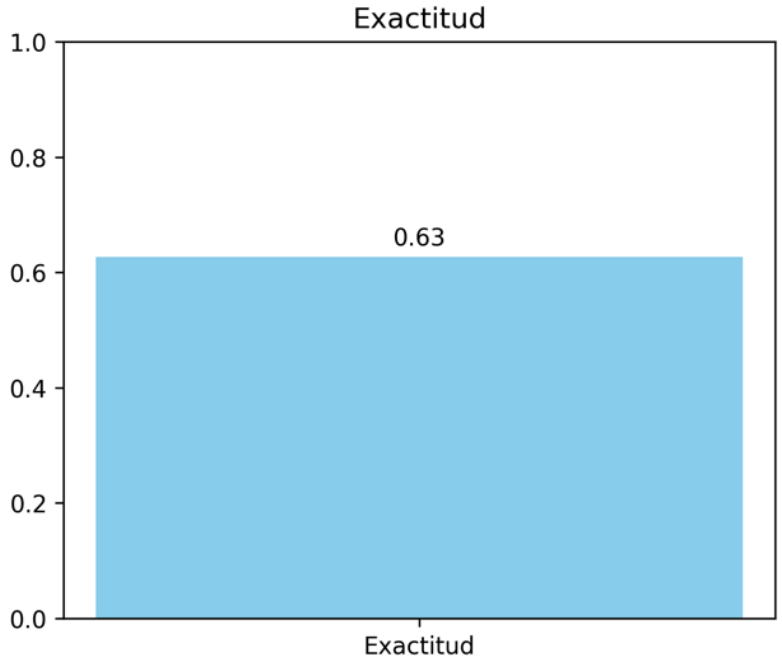
4. Experimentos

El modelo se entrenó con un subconjunto balanceado del conjunto de datos 20 Newsgroups, limitando a 100 documentos por categoría para garantizar una representación equitativa. Se utilizó una división de 75% para entrenamiento y 25% para prueba, con estratificación para mantener la distribución de categorías.

Se aplicaron las siguientes etapas:

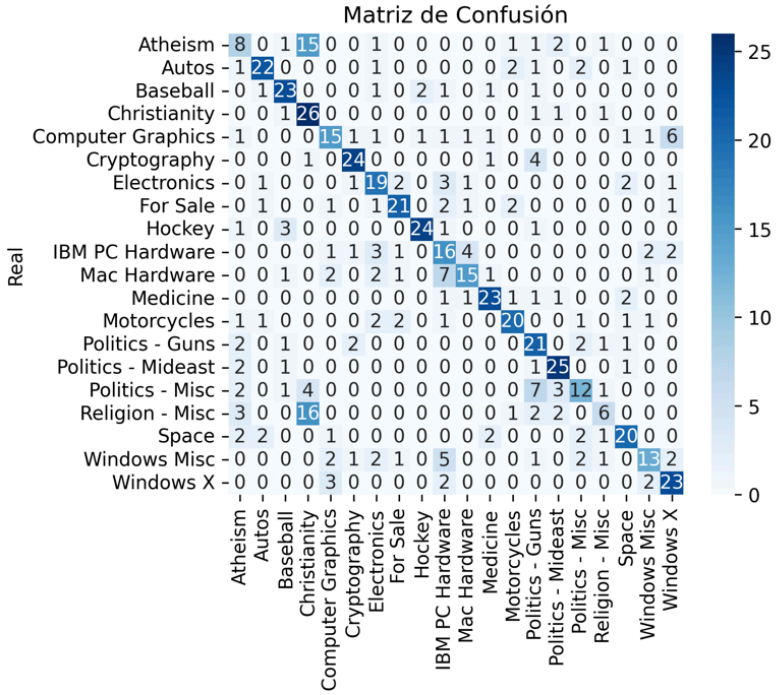
- Preprocesamiento: Limpieza de textos (eliminación de encabezados, URLs, caracteres especiales), tokenización, eliminación de stopwords y stemming con NLTK.
- Extracción de características: Vectorización TF-IDF con TfidfVectorizer de scikit-learn, configurado con max_features=20000, ngram_range=(1, 2), min_df=1, y max_df=1.0.
- Entrenamiento: Uso de MultinomialNB dentro de un Pipeline para integrar vectorización y clasificación.
- Evaluación: Cálculo de precisión, informe de clasificación (precisión, recall, F1-score) y matriz de confusión.

Los resultados muestran una precisión de 63.83% de exactitud. Se generaron visualizaciones, incluyendo la exactitud y la matriz de confusión para todas la categorías



5. Análisis

Los resultados se evidencian en las siguiente gráficas



La matriz de confusión mostró que categorías similares, como “Religion-Misc” y “Christianity”, tienden a confundirse, lo que refleja la dificultad de distinguir temas relacionados.

Los gráficos generados proporcionaron datos adicionale, estos se pueden visualizar al ejecutar el código adjunto:

- Precisión (VPP) = TP/(TP+FP) → Mide la proporción de casos positivos predichos que el modelo clasifica correctamente como positivos.
- Sensibilidad (Recall) = TP/(TP+FN) → Mide la proporción de casos positivos reales que el modelo clasifica correctamente como positivos.
- Valor Predictivo Negativo (VPN) = TN/(TN+FN) → Mide la proporción de casos negativos predichos que el modelo clasifica correctamente como negativos.
- Especificidad = TN/(TN+FP): Mide la proporción de casos negativos reales que el modelo clasifica correctamente como negativos
- Medida de Jaccard = TP/(TP+FP+FN) → Mide la proporción de casos positivos correctos respecto al total de casos positivos predichos y reales.

6. Conclusiones

El clasificador Naive Bayes con TF-IDF demostró ser efectivo para la clasificación de tópicos en noticias periodísticas, logrando una precisión aceptable. Sin embargo, la confusión entre categorías similares indica que técnicas avanzadas, como el uso de bigramas o modelos más complejos, podrían mejorar el rendimiento. Este sistema tiene un gran potencial para aplicaciones periodísticas, facilitando la organización de contenido.

La experiencia permitió profundizar en el uso de Naive Bayes y TF-IDF en tareas de clasificación de texto.

7. Código fuente

El código fuente, incluido en el archivo codigo.txt, está documentado y abarca:

- Carga y preprocesamiento del conjunto de datos 20 Newsgroups.
- Vectorización TF-IDF y entrenamiento del modelo Naive Bayes.
- Implementación de una GUI con Tkinter para interacción y visualización.
- Generación de métricas y gráficos de análisis.

8. Referencias

[1] Wikipedia. (2023). Naive Bayes classifier. Recuperado de https://en.wikipedia.org/wiki/Naive_Bayes_classifier.

[2] Wikipedia. (2023). Tfidf. Recuperado de https://en.wikipedia.org/wiki/Tfidf.

[3] scikit-learn. (2019). 5.6.2. The 20 newsgroups text dataset. Recuperado de https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html.

[4] GeeksforGeeks. (2025). Naive Bayes Classifiers. Recuperado de https://www.geeksforgeeks.org/machine-learning/naive-bayes-classifiers/.

[5] Analytics Vidhya. (2025). Naive Bayes Classifier Explained With Practical Problems. Recuperado de https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/.