Universidad de los Andes
Docente: Haydemar Nuñez
Curso: ISIS-3301

Proyecto 1- Entrega 1

Integrantes:

Bryan Orjuela Melo - 202112346

Daniel Monzon Arias - 202012345

Juan David Vasquez - 201914782

1. Entendimiento del Negocio y Enfoque Analítico

El auge de las redes sociales y la digitalización de la información han facilitado la difusión masiva de noticias falsas, impactando la percepción pública y la toma de decisiones. En este contexto, nuestro proyecto busca desarrollar un modelo de aprendizaje automático capaz de detectar noticias falsas dentro del ámbito político, permitiendo a medios de comunicación y entidades gubernamentales mitigar la desinformación.

Desde una perspectiva analítica, el problema se enmarca dentro de la analítica predictiva, dado que nuestro objetivo es anticipar si una noticia es real o falsa con base en sus características textuales. El aprendizaje aplicado es supervisado, dado que contamos con un conjunto de datos etiquetado. La tarea de aprendizaje corresponde a clasificación, ya que la variable objetivo es binaria (Real o Falsa). Los modelos considerados incluyen Naïve Bayes, Árboles de Decisión y Clasificador por Gradiente Estocástico, dado su desempeño comprobado en problemas de análisis de texto.

Se ha generado un Canvas con la información relevante para el desarrollo del proyecto. Este se encuentra disponible dentro de la wiki en el repositorio en GitHub.

2. Entendimiento y Preparación de los Datos

2.1. Entendimiento de los Datos

Para este estudio, utilizamos un conjunto de datos compuesto por 57,063 registros extraídos de distintos medios digitales. Las principales columnas del dataset son:

- ID: Identificador único de cada noticia.
- Label: Indica si la noticia es real (0) o falsa (1).
- Título: Encabezado de la noticia.
- Descripción: Resumen o fragmento del contenido de la noticia.

- Fecha: Momento en el que fue publicada.

Al realizar el perfilamiento de los datos, identificamos 16 valores nulos en la columna "Título" y 35,323 valores nulos en "Fecha", los cuales no afectan el análisis y fueron ignorados. También detectamos 450 registros duplicados, los cuales fueron eliminados para garantizar la integridad del dataset.

En términos de distribución, observamos que el 42.16% de las noticias eran reales y el 57.84% falsas, lo que sugiere un leve desbalanceo que podría afectar el rendimiento de los modelos.

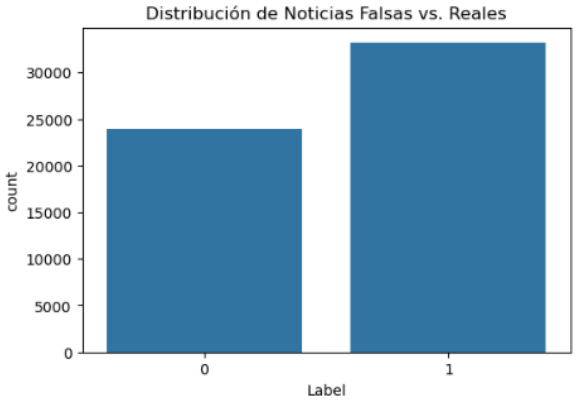

Figura 1. Distribución de noticias falsas respecto a noticias reales.

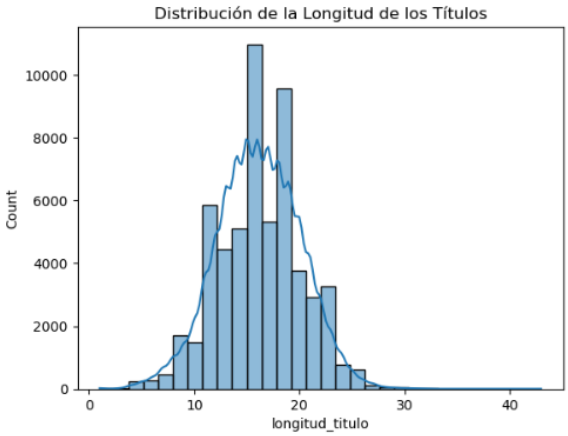

Figura 2. Distribución de la longitud de los títulos en las diferentes noticias.

Figura 3. Nube de palabras en los títulos de las noticias.

2.2. Preparación de los Datos

Para la representación de texto, utilizamos la técnica TF-IDF (Term Frequency-Inverse Document Frequency), seleccionando las 10,000 palabras más relevantes y transformando los textos en vectores numéricos. Además, realizamos los siguientes procesos de limpieza:

- Conversión a minúsculas.
- Eliminación de caracteres especiales y números.
- Eliminación de palabras vacías (stopwords).
- Aplicación de lematización para reducir las palabras a su forma base.

Los datos fueron divididos en 80% para entrenamiento y 20% para prueba, asegurando que la distribución de clases se mantuviera equilibrada en ambas particiones.

```
Distribución en el conjunto de entrenamiento:
Clase 0: 19088 registros (42.16%)
Clase 1: 26189 registros (57.84%)

Distribución en el conjunto de prueba:
Clase 0: 4772 registros (42.16%)
Clase 1: 6548 registros (57.84%)

Dimensiones del conjunto de entrenamiento: (45277, 14000)
Dimensiones del conjunto de prueba: (11320, 14000)
```

Figura 4. Distribución de los datos en los conjuntos de entrenamiento y prueba.

	ID	Label	Titulo	Descripcion	Fecha
0	ID	1	the guardi va sanchez europ necesit	diari britan public pas juev editorial	2023-02-06 00:00:00
1	ID	0	revel gobiern negoci liber mirel cam	revel gobiern negoci liber mirel cam	2023-01-10 00:00:00
2	ID	1	ahor nunc joan fust estatut valenciar	valencian convoc castell fiest grand	*Missing value*
3	ID	1	iglesi alient yoland diaz erc eh bildu	polit igual negoci empresari negoci	2022-03-01 00:00:00
4	ID	0	puigdemont ningun tragedi repeticio	entrev punt avui lid jxcat desdramati	2018-09-03 00:00:00

Figura 5. Visualización de una muestra de los datos después del proceso de preparación.

3. Modelado y Evaluación

3.1. Modelo 1 - Naïve Bayes (Bryan Orjuela Melo)

Naïve Bayes es un modelo probabilístico basado en la independencia condicional de las palabras dentro de un texto. Su simplicidad y rapidez lo hacen una opción eficiente para clasificación de texto. Se optimizó el hiperparámetro alpha, obteniendo el mejor desempeño con un valor de 0.01.

- Precisión: 90.0%
- Recall: 89.0%
- F1-score: 89.0%
- Exactitud: 89.9%

```
Alpha: 0.01 - Exactitud: 0.8994
Alpha: 0.1 - Exactitud: 0.8969
Alpha: 0.5 - Exactitud: 0.8920
Alpha: 1 - Exactitud: 0.8895
Alpha: 2 - Exactitud: 0.8843
Alpha: 5 - Exactitud: 0.8727
```

Figura 6. Resultado de exactitud del modelo Naïve Bayes bajo optimización del hiperparámetro alpha.

```
Mejor valor de alpha: 0.01 con exactitud de 0.8994

Reporte de Clasificación con el Mejor Alpha:
              precision    recall  f1-score   support

           0       0.95      0.80      0.87      4772
           1       0.87      0.97      0.92      6548

    accuracy                           0.90     11320
   macro avg       0.91      0.89      0.89     11320
weighted avg       0.90      0.90      0.90     11320
```

Figura 7. Métricas de calidad del modelo Naïve Bayes con hiperparámetro alpha que dio mejores resultados.

La matriz de confusión evidenció que el modelo cometía errores en la clasificación de noticias reales, lo que afectaba su rendimiento en escenarios donde minimizar falsos negativos es crucial.

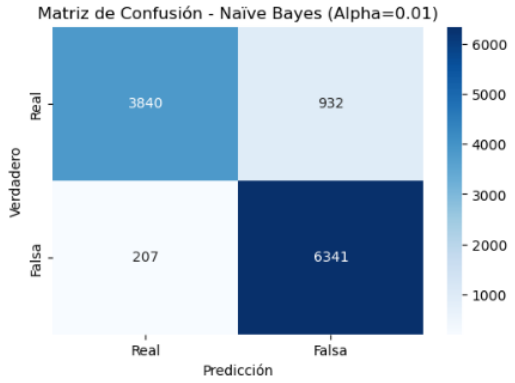

Figura 8. Matriz de confusión del mejor modelo Naïve Bayes.

3.2. Modelo 2 - Árboles de Decisión (Daniel Monzón Arias)

Los Árboles de Decisión son modelos interpretables que funcionan dividiendo iterativamente los datos en función de ciertas reglas. Inicialmente, obtuvimos un sobreajuste severo, con una exactitud del 100% en entrenamiento y 92.7% en prueba. Sin embargo, tras optimizar parámetros como la profundidad del árbol y el número mínimo de muestras por nodo, logramos mejorar su generalización.

- Precisión: 93.0%
- Recall: 92.0%
- F1-score: 92.0%
- Exactitud (Optimizado): 92.2%

Classification Report:

	precision	recall	f1-score	support
0	0.94	0.88	0.91	4772
1	0.92	0.96	0.94	6548
accuracy			0.93	11320
macro avg	0.93	0.92	0.92	11320
weighted avg	0.93	0.93	0.93	11320

Figura 8. Métricas de calidad del modelo con Árboles de Decisión (optimizado).

Modelo Inicial

```
Exactitud en entrenamiento: 1.000
Exactitud en prueba: 0.927
```

Modelo Optimizado

```
Exactitud en entrenamiento: 0.993
Exactitud en prueba: 0.929
```

Figura 9. Comparación de la exactitud entre el modelo inicial y el modelo optimizado.

El modelo mostró un buen equilibrio en la clasificación de ambas clases, pero su tiempo de entrenamiento y complejidad fueron mayores en comparación con otros modelos. Adicionalmente, por los resultados arrojados parece que ha ocurrido overfitting sobre los datos del modelo, siendo que se desempeña muy bien (cerca de lo ideal) sobre los datos de entrenamiento, pero la exactitud disminuye en alrededor de un 10% una vez se prueba con datos “reales”.

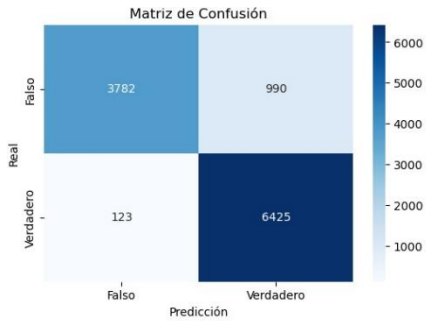

Figura 10. Matriz de confusión del modelo con Árboles de Decisión (optimizado).

3.3. Modelo 3 - Clasificador por Gradiente Estocástico (Juan David Vásquez)

Este modelo es una variante de regresión logística optimizada mediante gradientes estocásticos, permitiendo entrenar de manera eficiente en grandes volúmenes de datos y además aplicable al problema de clasificación. Tras ajustar hiperparámetros como la tasa de aprendizaje y el número de iteraciones, logramos un desempeño robusto:

- Precisión: 90.9%
- Recall: 90.1%
- F1-score: 89.9%
- Exactitud: 90.17%

	precision	recall	f1-score	support
0	0.9674	0.7967	0.8738	4772
1	0.8687	0.9805	0.9212	6548
accuracy			0.9030	11320
macro avg	0.9181	0.8886	0.8975	11320
weighted avg	0.9103	0.9030	0.9012	11320

Figura 11. Métricas de calidad del modelo con Clasificador de Gradiente Estocástico.

```
Accuracy train: 0.9204673454513329
Accuracy test: 0.9030035335689046
```

Figura 12. Comparación de la exactitud del modelo entre los datos de entrenamiento y prueba.

Se observan en general buenos resultados que no se encuentran muy lejanos entre sí para los conjuntos de entrenamiento y prueba por lo cual se considera que el modelo podría generalizar de forma apropiada.

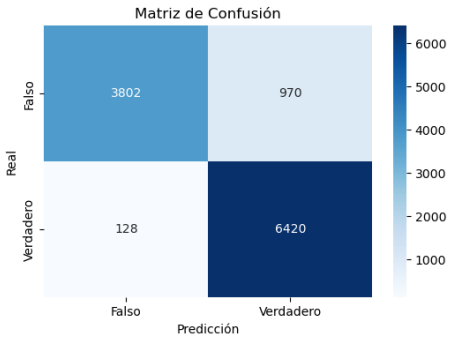

Figura 13. Matriz de confusión del modelo con Clasificador de Gradiente Estocástico.

También se realizó una prueba para entrenar el modelo con unos datos donde se combinara el título con la descripción de la noticia, sin embargo esta arrojó una precisión de 0.899 (menor a la del modelo inicial) los cual se asume se debe a la simplificación en la representación de las noticias y la pérdida de la importancia del título. Dado que nuestro objetivo era minimizar la clasificación errónea de noticias falsas como reales, el Clasificador por Gradiente Estocástico resultó ser la mejor opción por su balance entre precisión y recall.

4. Resultados

4.1. Comparación y Selección del Modelo Final

Al comparar las métricas de los modelos, observamos lo siguiente:

- Naïve Bayes: Mayor rapidez, pero menor desempeño en recall.
- Árboles de Decisión: Mayor exactitud, pero con tendencia al sobreajuste.
- Clasificador por Gradiente Estocástico: Mejor balance entre precisión y recall.

Dado nuestro objetivo de minimizar falsos negativos (clasificar noticias falsas como reales), seleccionamos Árboles de Decisión como nuestro modelo final optimizado, logrando una exactitud de 92.2% y un buen equilibrio en las métricas de clasificación.

4.2. Análisis de Palabras Clave

Analizando los términos más influyentes en las clasificaciones, encontramos que las noticias falsas solían contener palabras asociadas a eventos políticos y nombres propios, como: "pp", "vox", "cas", "sanchez", “psoe”. Por otro lado, las noticias reales mostraban términos más neutrales como "canari", "per cataluny", "inici vers". Esto se observó en los diferentes modelos entrenados para el problema de clasificación.

```
Número de características en el vectorizador: 14000
Palabras más representativas de noticias reales:
canari, equ, per cataluny, inici vers, vers per, vers, gobiern, inici, per, cataluny

Palabras más representativas de noticias falsas:
madr, vox, cas, pso, pod, sanchez, gobiern, pp
```

4.3. Archivo de Predicciones

Como parte de la entrega, generamos un archivo CSV con las predicciones realizadas sobre el conjunto de prueba, facilitando la comparación con los modelos de otros grupos.

4.4. Video y Documento

Preparamos un video de 5 minutos detallando el proceso, los resultados obtenidos y el impacto del modelo. Además, entregamos el presente documento con la información completa del proyecto.

5. Trabajo en Grupo

5.1) Roles y Tareas Realizadas

Para el desarrollo del proyecto, cada integrante asumió un rol específico para garantizar la correcta ejecución de las tareas y el cumplimiento de los objetivos.

- Bryan Orjuela Melo (202112346) - Líder de Proyecto y Modelo Naïve Bayes
    - Gestionó el cronograma del proyecto y organizó reuniones de avance.
    - Supervisó la distribución equitativa de las tareas y se encargó de subir la entrega.
    - Implementó y evaluó el modelo de Naïve Bayes, analizando su rendimiento.
    - Documentó la primera versión de los resultados obtenidos con este modelo.
    - Retos enfrentados: Ajuste del parámetro alpha para mejorar rendimiento.
    - Solución: Pruebas con distintos valores de alpha y análisis de métricas.
- Daniel Monzón Arias (202012345) - Líder de Datos y Modelo de Árboles de Decisión
    - Se encargó del procesamiento y limpieza de los datos, eliminando valores nulos y duplicados.
    - Implementó Árboles de Decisión, ajustando hiperparámetros para mejorar su desempeño.
    - Identificó problemas de sobreajuste y trabajó en estrategias de optimización.
    - Retos enfrentados: Alto tiempo de entrenamiento del modelo y sobreajuste.
    - Solución: Reducción de profundidad y ajuste del criterio de división de nodos.
- Juan David Vásquez (202112345) - Líder de Analítica y Modelo de Gradiente Estocástico
    - Evaluó métricas de rendimiento para la selección del mejor modelo.
    - Implementó y afinó el Clasificador por Gradiente Estocástico, optimizando su desempeño.
    - Comparó resultados entre modelos y justificó la elección del modelo final.
    - Retos enfrentados: Variabilidad en los resultados y sensibilidad a los hiperparámetros.
    - Solución: Ajuste de tasas de aprendizaje y regularización para mejorar estabilidad.

Cada integrante dedicó aproximadamente 15 horas al desarrollo del proyecto, distribuidas en exploración de datos, implementación de modelos y documentación.

5.2) Uso de ChatGPT en el Proyecto

Utilizamos ChatGPT como apoyo en distintas etapas del proyecto, principalmente en:

- Análisis y justificación de métricas de evaluación de los modelos.
- Redacción de documentos técnicos y guiones de presentación.
- Resolución de errores en la implementación y optimización del código.

5.3) Distribución de Puntos y Evaluación del Trabajo en Equipo

Dado que todos los integrantes contribuyeron equitativamente al desarrollo del proyecto, se ha decidido distribuir los 100 puntos de la siguiente manera:

- Bryan Orjuela Melo - 33 puntos
- Daniel Monzón Arias - 34 puntos
- Juan David Vásquez - 33 puntos

La distribución refleja una carga de trabajo equilibrada y reconoce la importancia de cada contribución al éxito del proyecto.

- Reuniones Realizadas

Para garantizar un buen desempeño en el desarrollo del proyecto, llevamos a cabo varias reuniones:

1. Reunión de lanzamiento y planeación
    - Se definieron los roles y se estableció la metodología de trabajo.
    - Se realizó una lluvia de ideas sobre los modelos a utilizar.
2. Reunión de ideación
    - Se exploraron los datos y se definió el enfoque del problema.
    - Se seleccionaron las técnicas de preprocesamiento y vectorización.
3. Reuniones de seguimiento (semanales)
    - Revisión del avance en la implementación de cada modelo.
    - Evaluación de métricas y ajustes en los hiperparámetros.
4. Reunión de finalización
    - Consolidación de resultados y selección del modelo final.
    - Preparación de la presentación y revisión de entregables.

5.4) Puntos a Mejorar para la Siguiente Entrega

A lo largo del proyecto, identificamos algunos aspectos que podemos mejorar en futuras entregas:

- Optimización del tiempo de entrenamiento en modelos más complejos.
- Exploración de técnicas avanzadas de NLP, como embeddings de palabras.
- Automatización del preprocesamiento de datos para mayor eficiencia.
- Mejor distribución del tiempo para evitar sobrecarga en la última fase del proyecto.