

UNIVERSIDAD DE LOS ANDES
DEPARTAMENTO DE INGENIERÍA DE
SISTEMAS Y COMPUTACIÓN



PROYECTO: ETAPA 1

ISIS 3301 – Inteligencia de Negocios

Andres Ortiz ()
Integrante 2 codigo
Integrante 3 codigo

2025

Tabla de contenido:

Introducción.....	3
Objetivo.....	3
1. Documentación del proceso de aprendizaje automático.....	4
2. Entendimiento y preparación de los datos.....	5
2.1. Calidad de texto:.....	6
2.2. Tokenización:.....	6
3. Modelado y evaluación.....	6
3.1. Objetivo y tarea de aprendizaje.....	6
5. Trabajo en equipo.....	7













Introducción

La UNFPA (United Nations Population Fund) adoptó la Agenda 2030 para Desarrollo Sostenible que trata de erradicar la pobreza, asegurar acceso igual a la salud, promover igualdad de género, y reducir impacto ambiental, entre otras iniciativas. La agenda estableció 17 Objetivos de Desarrollo Sostenible (ODS) y 169 metas asociadas. Dentro de esta estructura, se monitorea y se evalúan las políticas públicas por medio de herramientas de participación ciudadana para identificar asuntos y evaluar soluciones en relación con estos ODS. Un reto grande en este proceso es el análisis de grandes volúmenes de texto de las encuestas ciudadanas porque requieren recursos significativos y conocimiento de expertos.

Objetivo

El objetivo de este proyecto es ayudar a la UNFPA a desarrollar una aplicación analítica capaz de categorizar las opiniones de la ciudadanía en tres ODS: **1 (Fin de la pobreza)**, **2 (Salud y bienestar)**, y **3 (Educación de calidad)**. El modelo debe ser integrado en una aplicación web o móvil y que pueda ser utilizada de forma continua y que sea re-entrenable con nuevos textos naturales.

1. Documentación del proceso de aprendizaje automático

TAREA DE APRENDIZAJE  <p>Tipo de tarea: Aprendizaje supervisado de clasificación</p> <p>Predicción: Clasificar opiniones de ciudadanos de acuerdo a los términos ODS (tipo 1, tipo 3, y tipo 4) relacionados con "Fin de la pobreza", "Salud y bienestar" y "Educación de calidad"</p> <p>Resultados: La categorización de cada opinión en uno de los términos ODS</p> <p>Observar: El resultado se obtiene en segundos (dependiendo de la cantidad de datos)</p>	DECISIONES  <p>Las predicciones del modelo permiten generar recomendaciones que vinculan opiniones de la ciudadanía con políticas correspondientes a cada término ODS. Esto puede facilitar la toma de decisiones en la planificación de estrategias para atacar las iniciativas.</p>	PROPUESTA DE VALOR  <p>El beneficiario final es la UNFPA y las organizaciones que trabajan para lograr las metas de la Agenda 2030. El problema es que los datos proporcionados por los ciudadanos son muy grandes lo que causa que el análisis y clasificación manual requiere muchos recursos y expertos especializados. El modelo puede automatizar esta labor y reducir el tiempo de análisis considerablemente pero siempre habrá riesgo de que algunas opiniones no queden clasificadas correctamente.</p>	RECOLECCIÓN DE DATOS  	FUENTES DE DATOS  <p>Los datos que se utilizarán en este proyecto vienen de opiniones textuales proporcionados por los ciudadanos respecto ODS 1,3, y 4.</p>
SIMULACIÓN DE IMPACTO  <p>La automatización del análisis textual reduce el tiempo de labor humano y ayuda a disminuir recursos ayudando a que se puedan destinar para otros propósitos. Sin embargo, el costo es que puedan haber clasificaciones erróneas por parte del modelo que pueden afectar la asignación de recursos públicos para este proyecto. Para medir el éxito del modelo, usaremos las métricas F1-score ya que refleja equilibrio entre precisión y recall.</p>	APRENDIZAJE (USO DEL MODELO)  <p>Si tenemos en cuenta que el ciudadano proporciona su respuesta y que esa sea su única función, podríamos decir que el usuario final en realidad es el funcionario y no el ciudadano. En este caso, el modelo utilizará procesamiento por lotes para poder entregar informes agregados. El modelo se puede ejecutar periódicamente ya que la clasificación en tiempo real de cada respuesta no es una prioridad dado que las decisiones del gobierno no se basan en un solo texto sino en tendencias agregadas de varios ciudadanos. El uso sería por medio de una aplicación web donde ciudadanos tendrían una ventana de tiempo (tal vez en una escala de tiempo semanal)</p>		CONSTRUCCIÓN DE MODELOS  <p>Este proyecto hará uso de 3 algoritmos diferentes en la fase experimental para clasificar texto:</p> <ol style="list-style-type: none"> 1. Regresión logística 2. Máquina de soporte vectorial 3. Naive Bayes <p>En la app final, solo uno será seleccionado haciendo una evaluación comparativa entre el mejor F1-score.</p> <p>Para generar el modelo toca procesar datos, vectorizar, entrenar, y comparar. Este tiempo está en escala de días pero en un escenario real no debería superar una escala de horas.</p>	INGENIERÍA DE CARACTERÍSTICAS  <p>Para poder analizar los textos toca realizar procesos de tokenización, lematización, y eliminación de stop words. Después de este preprocesamiento toca vectorizar el texto con TF-IDF. Con esto podemos hacer el análisis de palabras más influyentes categorizadas por ODS.</p>
	MONITOREO 			

2. Entendimiento y preparación de los datos

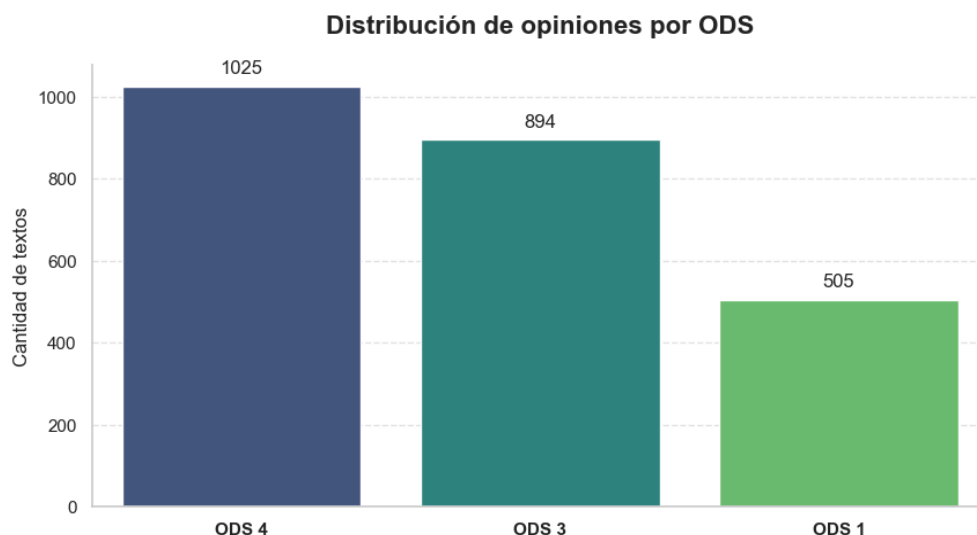
Se inició importando librerías como pandas, seaborn, matplotlib, nltk, re, spacy, sklearn, imblearn, entre otras, para procesos como la creación de gráficos, procesamiento de lenguaje, preparación, modelamiento y evaluación del modelo. Lo siguiente fue definir la ruta del archivo, donde se hizo una lectura de la misma con un manejo de errores, usando try para importar el archivo si la ruta es correcta y except para detectar e imprimir un mensaje de advertencia si la ruta no lo es. Después de eso se incluye un if para imprimir los primeros 5 datos del dataset e información parcial con la función info, a partir de lo cual se deduce lo siguiente:

Tamaño del dataset: 2424

Total de variables: 2

- **textos (object):** Opiniones de la ciudadanía
- **labels (int64):** Objetivos de Desarrollo Sostenible u ODS (*1: Fin de la pobreza, 3: Salud y bienestar, 4: Educación de calidad*)

Con las librerías de seaborn y matplotlib se creó un gráfico para observar la distribución de los textos por objetivo.



Se observa como el objetivo 4, relacionado con educación, tiene asociados la mayor cantidad de textos, más de 1000, que representa el 42.2% de todos los textos recolectados para el proyecto. Mientras que el objetivo 1, relacionado con el fin de la pobreza, fue el que menos textos tenía. Esto propone un desbalance en las clases de, por lo que más adelante se puede ser problemático para la clasificación.

2.1. Calidad de texto:

De entrada se observó dos aspectos de la calidad de datos, la nulidad y la duplicidad, con las funciones `isnull` y `duplicate` se hizo un conteo de los datos nulos y duplicados tanto en conjunto como por columna, donde no se encontró ninguna anomalía.

2.2. Tokenización:

Se inició descargando la lista de stopwords, donde posteriormente se creó la función `limpiar_texto` para hacer los siguientes cambios: pasar todo el texto a minúscula, eliminar signos de puntuación, tildes, espacios repetidos. También con la función `token.lenma_` se incluyó la lematización de los textos omitiendo espacios, stopwords y que los textos tengan más de la palabra. Imprimiendo un nuevo dataset con 4 columnas (`labels`, `texto`, `texto_limpio`, `tokens`) y se construyó un nuevo dataset con base en la columna `tokens` para redistribuir la

Eliminación de signos de puntuación, números, stopwords y otros elementos que podrían .

3. Modelado y evaluación

3.1. Objetivo y tarea de aprendizaje

El objetivo de estos modelos es realizar tareas de clasificación sobre las opiniones de los ciudadanos en las tres diferentes categorías ODS. Las tareas son supervisadas, ya que partimos desde un set de datos cuyos labels se conocen. → La salida de las tareas es un label discreto {1, 3, 4} (mencionado en la sección previa).

3.2 Búsqueda de hiperparámetros

Para optimizar el desempeño de los modelos, aplicamos `GridSearchCV` con 5-fold cross-validation y escogimos F1-weighted como la métrica de puntuación para cada

modelo. Esto aseguró que la búsqueda tuviera un balance entre precisión y recall entre las categorías. Además, cada pipeline hizo uso de SMOTE para manejar categorías minoritarias.

Por simplicidad, solamente exploramos un hiperparámetro para cada modelo:

- **Regresión Logística:** C (inversa de regularization strength). Controla que tanto el modelo penaliza coeficientes grandes. Un valor bajo de C implica regularización más fuerte para un modelo más simple con menos sobreajuste mientras que un valor alto de C implica regularización más débil para un modelo más flexible pero con más riesgo de sobreajuste. **Grid: [0.1, 1, 10, 100]**
- **Naïve Bayes:** α (Laplace smoothing). Determina qué tanta probabilidad se le agrega a palabras poco vistas en las respuestas de los ciudadanos. Un valor muy bajo de α corre el riesgo de sobreajuste y probabilidades nulas mientras que un valor muy alto de α causa que el modelo sea muy uniforme causando la pérdida de poder predictivo. Para este proyecto, este modelo requiere un α ajustado tal que balancea palabras exóticas sin aplastar tanto la frecuencia de palabras comunes. **Grid: [0.1, 0.5, 1.0]**
- **Linear Support Vector Classifier (LinearSVC):** C (parámetro de penalización de clasificación errónea). Este controla el beneficio entre maximizar el margen y permitir errores de clasificación. Un valor bajo de C permite un margen más grande, tiene mayor sesgo, pero mejor generalización. Un valor muy alto de C tiene un margen de error más pequeño, tiene menor sesgo, pero con riesgo de sobreajuste.

4. Resultados

5. Trabajo en equipo