ASIGNATURA: INTELIGENCIA DE NEGOCIOS
ESTUDIANTES: JUAN DAVID BRICEÑO, DANIEL CLAVIJO, CARLOS MEDINA
CÓDIGOS: 201812887, 202122209, 202112046
PROGRAMA DE ESTUDIOS: INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

Proyecto de Analítica de Textos: Clasificación de Objetivos de Desarrollo Sostenible (ODS)

Sección 1: Proceso de Automatización

1.1 Descripción del Proceso:

En este proyecto, hemos implementado un proceso automatizado para la clasificación de textos según los Objetivos de Desarrollo Sostenible (ODS) de las Naciones Unidas. El proceso abarca desde la preparación de datos hasta la implementación de una API accesible, pasando por la construcción y persistencia del modelo.

Preparación de Datos y Construcción del Modelo:

Utilizamos un enfoque de pipeline que integra varios componentes principales:

1. Antes de la vectorización, implementamos un paso crucial de preprocesamiento del texto. Este proceso incluye varias etapas diseñadas para limpiar y estandarizar el texto de entrada:

- Conversión a minúsculas: Uniformiza todo el texto para evitar distinciones basadas en mayúsculas.

- Limpieza de caracteres: Eliminamos caracteres especiales y números, conservando solo letras y espacios. Esto incluye la preservación de caracteres españoles como acentos y la letra 'ñ'.

- Tokenización: Dividimos el texto en palabras individuales o "tokens".

- Eliminación de stopwords: Removemos palabras comunes en español que no aportan significado sustancial a la clasificación.

- Lematización: Reducimos las palabras a su forma base o lema, lo que ayuda a unificar variantes de la misma palabra. Este preprocesamiento es fundamental para mejorar la calidad de los datos de entrada al modelo, reduciendo el ruido y estandarizando la información textual. Esto, a su vez, contribuye a una mejor performance del modelo de clasificación.

2. Vectorización de texto: Empleamos la técnica TF-IDF (Term Frequency-Inverse Document Frequency) para convertir el texto en una representación numérica que el modelo puede procesar.

3. Clasificación: Implementamos un clasificador Naive Bayes, conocido por su eficacia en tareas de clasificación de texto. Este pipeline nos permite automatizar la transformación de texto crudo en predicciones de ODS de manera eficiente y reproducible.

1.2 Persistencia del Modelo:
Para garantizar la reutilización eficiente del modelo entrenado, implementamos un sistema de persistencia. El modelo se guarda en un archivo después del entrenamiento, lo que permite cargarlo rápidamente para futuras predicciones sin necesidad de reentrenarlo cada vez.

Acceso mediante API:

Desarrollamos una API REST utilizando el framework FastAPI, que proporciona dos endpoints principales:

1. Endpoint de Predicción: Recibe una o más instancias de texto y devuelve predicciones de ODS junto con probabilidades asociadas para cada instancia.

2. Endpoint de Reentrenamiento: Acepta nuevos datos de entrenamiento y reentrena el modelo actualizándolo al archivo persistente. Además, proporciona métricas de rendimiento (Precision, Recall, F1-score) después del reentrenamiento.

1.3 Implementación de Endpoints:

Nuestra API implementa los endpoints requeridos, asegurando que:

El endpoint de predicción pueda manejar múltiples instancias de texto y devolver resultados para cada una.
El endpoint de reentrenamiento procese nuevos datos, actualice el modelo y proporcione métricas de rendimiento. Los datos de entrada y salida se manejan en formato JSON, respetando el esquema del CSV original del proyecto.

1.4 Definiciones de Reentrenamiento:
Implementamos tres estrategias distintas de reentrenamiento:

1. Reemplazo Completo:

Descripción: Sustituye el modelo existente por uno nuevo entrenado solo con los datos más recientes.
Ventaja: Rapidez y simplicidad en la implementación.
Desventaja: Potencial pérdida de información valiosa de datos históricos.

2. Concatenación de Datos:

Descripción: Combina los datos nuevos con los existentes para entrenar un modelo actualizado.
Ventaja: Conserva el conocimiento previo mientras incorpora nueva información.
Desventaja: Puede ser computacionalmente costoso con grandes volúmenes de datos acumulados.

3. Ponderación de Datos:

Descripción: Asigna mayor peso a los datos nuevos durante el proceso de entrenamiento.

Ventaja: Permite un equilibrio ajustable entre la información histórica y la más reciente.

Desventaja: Requiere una cuidadosa calibración de los pesos para optimizar resultados.

Conclusión: Para este proyecto, elegimos implementar la estrategia de concatenación de datos como nuestro método principal de reentrenamiento. Esta decisión se basa en su capacidad para mantener el conocimiento acumulado mientras se adapta a nueva información, lo cual es crucial en el contexto dinámico de los ODS.

ACLARACIÓN: Debido a que se implementaron 3 Endpoints distintos para el reentrenamiento con el fin de probar la eficacia de cada uno, se decidió dejar la implementación de cada uno y poner en el Front como métodos alternativos los otros métodos no escogidos para este proyecto. Esto, con el fin de que el científico

de datos pueda escoger el método de reentrenamiento si es que considera oportuno utilizar otro.

Sección 2: Desarrollo de la Aplicación y Justificación

2.1 Descripción del Usuario/Rol

Nuestra aplicación está diseñada para atender las necesidades de dos roles principales dentro de organizaciones enfocadas en desarrollo sostenible:

1. Analista de Datos:

    - Responsabilidades: Clasificar diversos tipos de documentos (informes, propuestas, artículos) según su relación con los ODS.

    - Uso de la aplicación: Utiliza la interfaz para obtener clasificaciones rápidas y precisas, junto con las probabilidades asociadas. La aplicación se utiliza de una oración o frase al tiempo para obtener su clasificación.

2. Científico de Datos:

    - Responsabilidades: Mantener y mejorar continuamente el modelo de clasificación.

    - Uso de la aplicación: Emplea la interfaz para reentrenar el modelo con nuevos conjuntos de datos y evaluar su rendimiento a través de métricas clave. La aplicación recibe un archivo Excel para su reentrenamiento.

2.2 Conexión con el Proceso de Negocio

La aplicación se integra en los procesos de análisis y toma de decisiones de organizaciones orientadas a los ODS:

1. Clasificación Eficiente de Documentos:

    - Facilita la categorización ordenada de proyectos, informes y propuestas según los ODS relevantes.

2. Mejora Continua y Adaptabilidad:

    - Los científicos de datos pueden actualizar regularmente el modelo.

    - Asegura que la clasificación se mantenga precisa y relevante a medida que evolucionan los contextos y las interpretaciones de los ODS.

2.3 Importancia de la Aplicación

1. Para el Analista de Datos:

- Automatiza y agiliza el proceso de clasificación, reduciendo significativamente el tiempo requerido.

- Minimiza errores humanos en la categorización.

- Proporciona probabilidades que apoyan la toma de decisiones en casos ambiguos.

2. Para el Científico de Datos:

- Ofrece una interfaz intuitiva para el reentrenamiento del modelo, simplificando un proceso técnico complejo.

- Proporciona métricas de rendimiento inmediatas, facilitando la evaluación rápida de las actualizaciones del modelo.

- Permite experimentar con diferentes estrategias de reentrenamiento, fomentando la innovación y mejora continua.

3. Para la Organización:

- Mejora la eficiencia en la alineación de proyectos e iniciativas con los ODS específicos.

- Facilita el seguimiento y reporte de actividades relacionadas con los ODS, crucial para la toma de decisiones.

- Apoya la toma de decisiones estratégicas basadas en una clasificación consistente y objetiva de textos relevantes.

- Potencia la capacidad de la organización para responder rápidamente a nuevas oportunidades o desafíos relacionados con los ODS.

2.4 Desarrollo de la Aplicación Web

Hemos desarrollado una aplicación web intuitiva y funcional utilizando tecnologías modernas:

1. Interfaz para Analistas de Datos:

- Permite la entrada fácil de texto para clasificación.

- Muestra claramente la predicción del ODS y la probabilidad asociada.

- Ofrece recomendaciones basadas en la confianza de la predicción, ayudando en la interpretación de resultados.

2. Interfaz para Científicos de Datos:

- Facilita la carga de nuevos conjuntos de datos para reentrenamiento.

- Ofrece opciones para seleccionar entre diferentes métodos de reentrenamiento.

- Presenta métricas de rendimiento detalladas tras cada proceso de reentrenamiento.

La aplicación se ha diseñado con un enfoque en la experiencia del usuario, asegurando una interfaz accesible tanto en entornos de escritorio. Esto permite a los usuarios realizar sus tareas de manera eficiente.

En resumen, nuestra aplicación no solo automatiza y agiliza procesos críticos relacionados con los ODS, sino que también proporciona una plataforma adecuada y fácil de usar para la toma de decisiones informadas y la mejora continua en el ámbito del desarrollo sostenible.

Sección 3: Video

El video se encuentra publicado en el padlet.

Sección 4: Trabajo en equipo

Roles:

- **Líder de proyecto:** Daniel Clavijo
- **Ingeniero de datos:** Daniel Clavijo, Juan David Briceño y Carlos Medina
- **Ingeniero de software responsable del diseño desde la aplicación y resultados:** Daniel Clavijo y Juan David Briceño
- **Ingeniero de software responsable de desarrollar la aplicación final:** Daniel Clavijo y Juan David Briceño

Las tareas realizadas se dividieron de la siguiente manera:

Daniel y Juan David implementaron tanto el frontend como el backend, incluyendo el API REST y el pipeline junto con los endpoints, el documento estuvo también a cargo de ellos. Una primera aproximación del pipeline y el API REST lo realizó Carlos Medina, así como el video requerido.

En cuanto a los horarios, toda la realización del proyecto le tomo al grupo aproximadamente 10 horas.

El reto principal del proyecto fue una buena construcción del pipeline y el API REST, la planeación establecida fue que un integrante del equipo implementara el API y el backend, mientras que los otros dos integrantes implementaban el front y su conexión con el API, la realidad fue imposible

integrar el API y nos vimos obligados a volver a implementar el API REST y el pipeline con el fin de integrarlo al frontend.

La distribución de los 100 puntos la resolvimos de esta manera:

Daniel Clavijo: 40 puntos

Juan David Briceño: 40 puntos

Carlos Medina: 20 puntos