Etapa 2 Proyecto Inteligencia de Negocios

Integrantes

Daniel Pedroza 202123283
Miguel Gomez 202122562
Pablo Martinez 202122937

Proceso de Automatización..... 3
Introducción a la Automatización de Modelos de Analítica de Textos.....3
Pipeline de Preparación de Datos y Construcción del Modelo..... 3
Implementación de la API REST para Predicción y Reentrenamiento..... 4
Definiciones de reentrenamiento..... 4
Desarrollo de la Aplicación y Justificación..... 5
Descripción del Usuario Final y su Rol en la Organización.....5
Conexión con el Proceso de Negocio y Beneficios Esperados.....5
Diseño de la Aplicación: Interfaz de Usuario y Funcionalidades.....5
Integración del Modelo Analítico en la Aplicación.....6
Interacción con el Modelo: Predicción y Visualización de Resultados.....6
Resultados.....6
Resumen de los Resultados Obtenidos.....6
Impacto de los Resultados para el Usuario Final.....10
Trabajo en Equipo..... 11

Proceso de Automatización

Introducción a la Automatización de Modelos de Analítica de Textos

Considerando el contexto de la aplicación, en el que se debe realizar un análisis de grandes volúmenes de datos textuales, se identificó la necesidad de automatizar los modelos de analítica de textos. Este proceso permite extraer información clave de los comentarios relacionados con los Objetivos de Desarrollo Sostenible (ODS) —salud, educación e igualdad de género—, lo que facilita el desarrollo de políticas públicas orientadas a mejorar la gestión de estas problemáticas globales. El modelo elegido fue SVM, que demostró métricas altamente efectivas para clasificar las opiniones en las distintas categorías de los ODS. Esta clasificación permite evaluar los comentarios de manera más rápida, lo que a su vez posibilita la implementación de métodos más precisos para mejorar las políticas públicas vigentes.

Pipeline de Preparación de Datos y Construcción del Modelo

El pipeline se desarrolló utilizando las librerías, pandas, numpy, scikit-learn, nltk y spacy para el procesamiento de texto. Este proceso nos permitió transformar los datos que se encontraban en lenguaje natural a un formato para el aprendizaje automático.

Para la preparación de datos se tuvieron que hacer pasos del preprocesamiento:

- Corrección de Codificación:
    - Los textos contenían caracteres mal codificados, por lo que se aplicó una función que reemplaza los caracteres erróneos con su equivalente adecuado, asegurandonos de la integridad del contenido, se hizo un diccionario con los errores que se hallaron y su forma de escrita correcta.
- Normalización del texto:
    - Para esto hicimos todo el texto a minúsculas y se eliminan la puntuación y los acentos para evitar diferencias que puedan confundir al modelo.
- Conversión de Números:
    - Los números en el texto se convierten a palabras utilizando la librería num2words para español, lo que facilita el análisis del contexto numérico en el texto.
- Eliminación de Stopwords:
    - Con nltk, se eliminan las palabras stopwords en español. Esto reduce el ruido en los datos y mejora la relevancia de las palabras clave.
- Lematización de Verbos:
    - Con spacy, se identifican y lematizan los verbos para unificar las formas verbales, aumentando la consistencia del texto y mejorando la precisión del modelo.

Ahora nosotros construimos el modelo de aprendizaje con estos pasos, en el que hacemos una vectorización y usamos el modelo SVM

- Vectorización con TF-IDF:
    - Para esto utilizamos TfidfVectorizer para transformar el texto en vectores numéricos, de esta manera pasamos los textos a una forma de cuantificar la relevancia de cada palabra en el contexto del documento.

- Clasificación con SVM:
    - Se implementa el modelo de Support Vector Machine con probabilidad activada, capaz de clasificar las opiniones entre los tres ODS de interés. Se entrenó el modelo con un dataset que divide el 70% de los datos para entrenamiento y el 30% para prueba.

Implementación de la API REST para Predicción y Reentrenamiento

La API REST se implementó utilizando Flask. Existen dos endpoints principales: uno para realizar predicciones y otro para reentrenar el modelo con nuevos datos. El endpoint /api/predict recibe un JSON con una lista de oraciones en español y devuelve una predicción de la categoría ODS para cada oración, junto con las probabilidades asociadas. El endpoint /api/retrain permite al usuario cargar un archivo Excel con datos de texto y etiquetas de categorías ODS, lo cual inicia el proceso de reentrenamiento del modelo. Después de entrenar el modelo, se persiste el nuevo modelo utilizando joblib y se genera un reporte de métricas que detalla el desempeño del modelo actualizado. Esta estructura de la API permite tanto la predicción en tiempo real como la actualización del modelo.

Definiciones de reentrenamiento

Cuando se habla del reentrenamiento del modelo, se pueden considerar diversas maneras de abordarlo. Una opción es el reentrenamiento total, que consiste en entrenar el modelo desde cero con todos los datos disponibles, lo cual asegura una coherencia completa del modelo con el conjunto de datos, pero es muy intensivo en términos de recursos y tiempo. Otra opción es el reentrenamiento incremental, que permite actualizar el modelo con datos nuevos sin perder completamente el conocimiento previo, lo que puede ser eficiente en términos de tiempo y recursos; sin embargo, este enfoque puede presentar discrepancias si los nuevos datos difieren significativamente de los originales. Finalmente, existe el reentrenamiento con datos recientes, en el que se entrena el modelo utilizando únicamente los datos más recientes proporcionados, olvidando los datos previos. Este enfoque es útil cuando los datos antiguos han perdido relevancia o cuando se requiere una rápida adaptación a cambios, aunque la pérdida de conocimiento previo puede afectar la precisión del modelo para el cliente.

Cada enfoque tiene sus ventajas y desventajas. El reentrenamiento total asegura que el modelo esté completamente alineado con todo el conjunto de datos, pero requiere más tiempo y recursos. El reentrenamiento con datos recientes permite una rápida adaptación a cambios recientes, pero al ignorar datos antiguos puede afectar la precisión. El reentrenamiento incremental equilibra eficiencia y uso de recursos, aunque puede no ser tan efectivo si los nuevos datos son muy diferentes a los antiguos.

Decisión Final sobre el Proceso de Reentrenamiento

La decisión final fue optar por el reentrenamiento con datos recientes, ya que permite a la organización encargada de los Objetivos de Desarrollo Sostenible (ODS) actualizar el modelo de manera rápida y eficiente cada vez que recibe nuevos datos. Este enfoque

facilita la actualización continua del modelo sin la necesidad de emplear una alta cantidad de recursos o tiempo, lo cual es ideal para una organización que maneja datos en constante cambio. Al utilizar únicamente los datos más recientes, el modelo se adapta a los contextos actuales de clasificación de textos relacionados con las tres ODS específicas, evitando posibles fallos de contexto que podrían surgir con el enfoque incremental. Además, esta opción permite que el modelo esté siempre alineado con la información más reciente, lo que resulta en una mayor precisión y mejora de los resultados generales, ayudando así a optimizar la toma de decisiones y el diseño de políticas públicas en áreas clave como educación, salud e igualdad de género.

Desarrollo de la Aplicación y Justificación

Descripción del Usuario Final y su Rol en la Organización

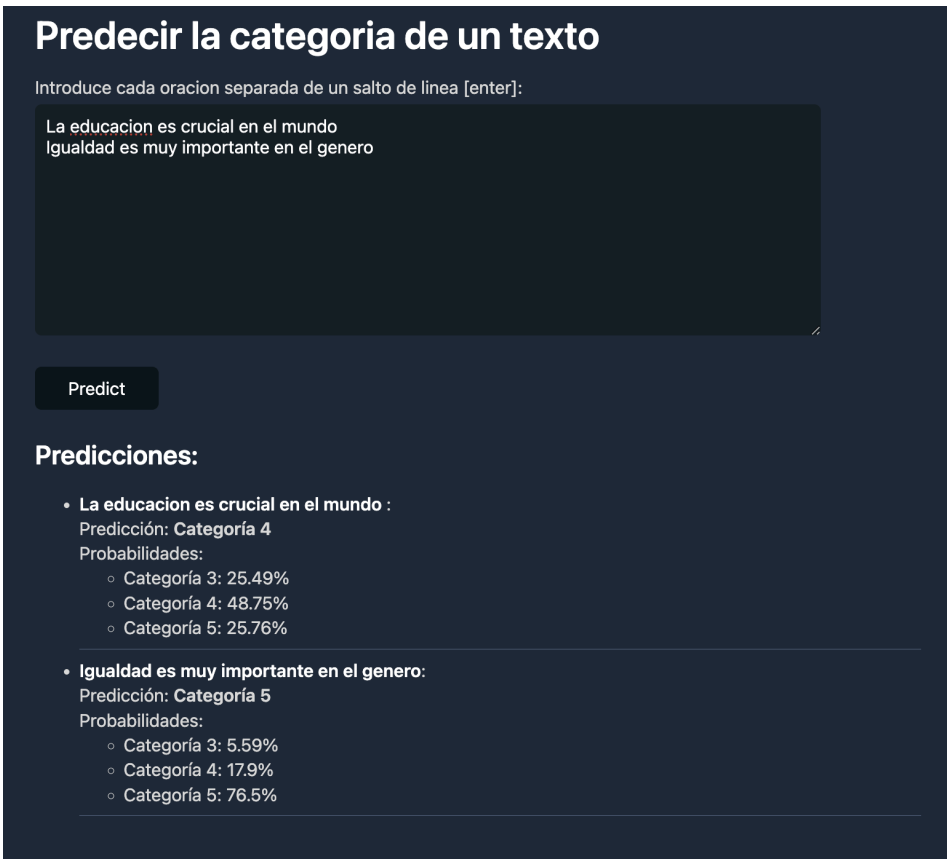
El usuario final es un analista de datos o tomador de decisiones en una organización que trabaja con los ODS. Su rol consiste en utilizar la aplicación para analizar textos, identificar temas clave, y tomar decisiones informadas con base en los resultados del análisis automatizado. Este usuario es clave para transformar los datos en información accionable que guíe las estrategias de la organización.

Conexión con el Proceso de Negocio y Beneficios Esperados

La aplicación se integra con los procesos de análisis de datos textuales, un área que suele requerir mucho tiempo y recursos humanos. Automatizar este análisis proporciona beneficios claros como la optimización de tiempos, la mejora en la precisión de los resultados, y la posibilidad de analizar grandes volúmenes de datos con menor esfuerzo. Esto, a su vez, permite que la organización tome decisiones más rápidas y fundamentadas.

Diseño de la Aplicación: Interfaz de Usuario y Funcionalidades

La interfaz de usuario está diseñada para ser intuitiva y fácil de usar, permitiendo al usuario cargar textos, recibir predicciones sobre su relación con los ODS junto con las probabilidades de cada predicción, y visualizar los resultados de manera clara. La aplicación incluye funcionalidades clave como el ingreso de textos y la visualización de las probabilidades asociadas a cada predicción. Del mismo modo permite entrenar el modelo con un dataset nuevo.



Integración del Modelo Analítico en la Aplicación

El modelo analítico, construido en la primera fase del proyecto, se integra en la aplicación para procesar automáticamente los textos ingresados por el usuario. El modelo utiliza técnicas de procesamiento de lenguaje natural para predecir la relación de los textos con los ODS y devolver una predicción con su probabilidad asociada. Esta integración facilita que el usuario obtenga resultados precisos sin necesidad de intervención manual.

Interacción con el Modelo: Predicción y Visualización de Resultados

El usuario interactúa con el modelo al ingresar textos y recibir predicciones automáticas sobre su relación con los ODS. La visualización de resultados es clara y detallada, mostrando las predicciones y las probabilidades asociadas. Esta interacción le permite al usuario evaluar rápidamente los datos procesados y utilizarlos para tomar decisiones informadas dentro de la organización.

Resultados

Resumen de los Resultados Obtenidos

La aplicación ofrece dos funcionalidades principales: el reentrenamiento del modelo y la predicción de categorías para textos específicos, la cual nos da unos resultados específicos

en cada caso. En el primer caso se permite un re entrenamiento completo del modelo en el cual, el usuario puede cargar un archivo en formato Excel con nuevos datos de entrenamiento. Una vez cargado, el modelo se entrena desde cero. Al finalizar el proceso, la aplicación muestra un reporte detallado de clasificación que incluye métricas como precisión, recall, F1-Score y soporte para cada una de las categorías. En este caso, el modelo tiene un desempeño consistentemente alto con una precisión, recall y F1-Score de 0.98 para las categorías 3, 4 y 5. Esto indica que el modelo es capaz de clasificar con alta exactitud los datos proporcionados.

Reentrenar el modelo con nuevos datos

Seleccionar Excel:

Choose File ODScat_345.xlsx

Reentrenar

Modelo reentrenado satisfactoriamente !

Classification Report:

Class	Precision	Recall	F1-Score	Support
3	0.98	0.99	0.98	372.0
4	0.98	0.98	0.98	402.0
5	0.99	0.99	0.99	441.0
Accuracy	0.98			
Macro Avg	0.98	0.98	0.98	1215.0
Weighted Avg	0.98	0.98	0.98	1215.0

Esta carga se logró con un volumen considerable de datos, aproximadamente 1,000 instancias distribuidas entre las categorías. Gracias a esto, los usuarios pueden tener una comprensión clara de cómo el modelo clasifica y evalúa el contenido, asegurando una categorización precisa y útil para el análisis posterior. Esto demuestra que la aplicación no solo es robusta, sino también escalable y adecuada para manejar grandes volúmenes de información, garantizando así la confiabilidad del sistema en diferentes escenarios.

El segundo resultado obtenido es mediante la segunda funcionalidad que es la predicción, ésta permite al usuario ingresar textos, los cuales son analizados y clasificados por el modelo en tiempo real.Cuando se agregan los textos, la aplicación muestra la categoría predicha de los ODS, junto con las probabilidades de pertenencia a cada categoría. Esto proporciona un nivel adicional de transparencia y permite al usuario entender la certeza del modelo para cada predicción. La aplicación logra esto mediante la clasificación de cada texto basándose en temas clave y palabras claves, permitiendo una rápida identificación y organización de contenidos según sus características principales.

Predecir la categoria de un texto

Introduce cada oracion separada de un salto de linea [enter]:

La salud en los pueblos indígenas es deficiente
El acceso a la educación en zonas rurales es limitado
Aumentan los casos de desnutrición infantil en la Guajira
Las escuelas carecen de infraestructura adecuada
Las mujeres en zonas rurales no reciben atención médica adecuada
El embarazo adolescente ha incrementado en el último año
Falta de personal docente en áreas apartadas
Los índices de analfabetismo son altos en comunidades indígenas
Se reportan casos de violencia de género en comunidades rurales
La cobertura de vacunación en la infancia es baja

Predict

Predicciones:

- **La salud en los pueblos indígenas es deficiente :**
  Predicción: **Categoría 3**
  Probabilidades:
  - Categoría 3: 99.99%
  - Categoría 4: 0.01%
  - Categoría 5: 0.0%

- **El acceso a la educación en zonas rurales es limitado :**
  Predicción: **Categoría 4**
  Probabilidades:
  - Categoría 3: 0.65%
  - Categoría 4: 99.09%
  - Categoría 5: 0.25%

- **Aumentan los casos de desnutrición infantil en la Guajira :**
  Predicción: **Categoría 3**
  Probabilidades:
  - Categoría 3: 87.72%
  - Categoría 4: 1.42%
  - Categoría 5: 10.86%

- **Las escuelas carecen de infraestructura adecuada :**
  Predicción: **Categoría 4**
  Probabilidades:
  - Categoría 3: 3.46%
  - Categoría 4: 96.08%
  - Categoría 5: 0.46%

- **Las escuelas carecen de infraestructura adecuada :**
  Predicción: **Categoría 4**
  Probabilidades:
  - Categoría 3: 3.46%
  - Categoría 4: 96.08%
  - Categoría 5: 0.46%

- **Las mujeres en zonas rurales no reciben atención médica adecuada :**
  Predicción: **Categoría 3**
  Probabilidades:
  - Categoría 3: 52.85%
  - Categoría 4: 0.01%
  - Categoría 5: 47.15%

- **El embarazo adolescente ha incrementado en el último año :**
  Predicción: **Categoría 3**
  Probabilidades:
  - Categoría 3: 70.33%
  - Categoría 4: 8.74%
  - Categoría 5: 20.93%

- **Falta de personal docente en áreas apartadas :**
  Predicción: **Categoría 4**
  Probabilidades:
  - Categoría 3: 0.36%
  - Categoría 4: 99.51%
  - Categoría 5: 0.13%

- **Los índices de analfabetismo son altos en comunidades indígenas :**
  Predicción: **Categoría 3**
  Probabilidades:
  - Categoría 3: 56.66%
  - Categoría 4: 39.87%
  - Categoría 5: 3.47%

- **Se reportan casos de violencia de género en comunidades rurales :**
  Predicción: **Categoría 5**
  Probabilidades:
  - Categoría 3: 0.0%
  - Categoría 4: 0.0%
  - Categoría 5: 100.0%

- **La cobertura de vacunación en la infancia es baja :**
  Predicción: **Categoría 3**
  Probabilidades:
  - Categoría 3: 84.37%
  - Categoría 4: 9.21%
  - Categoría 5: 6.42%

Impacto de los Resultados para el Usuario Final

El modelo de clasificación de textos para las ODS de salud, educación y equidad de género es una herramienta valiosa para quienes desarrollan políticas públicas. A través de esta aplicación, los usuarios pueden:

- Procesar grandes volúmenes de datos sin errores de contexto, facilitando el análisis de opiniones y preocupaciones ciudadanas.
- Identificar rápidamente temas prioritarios y áreas que requieren intervención urgente, permitiendo la formulación de políticas más efectivas y alineadas con las necesidades reales.
- Evaluar el nivel de confianza en la clasificación de cada categoría ODS mediante porcentajes de precisión, lo cual ayuda a tomar decisiones basadas en datos sólidos.

La aplicación, desplegada localmente, proporciona una interfaz intuitiva que simplifica el uso del modelo. A través de esta interfaz, los usuarios pueden:

- Ingresar y procesar textos de forma rápida, obteniendo de inmediato la clasificación correspondiente con el porcentaje de precisión.
- Usar la aplicación de manera sencilla, gracias a su diseño intuitivo, accesible para personas con distintos niveles de habilidad técnica.
- Reentrenar el modelo de manera sencilla con datos recientes, lo que mantiene las predicciones actualizadas y relevantes.

En conclusión, el modelo de clasificación de textos implementado proporciona una herramienta poderosa para mejorar la formulación de políticas públicas en Colombia en torno a las ODS de salud, educación y equidad de género. Gracias a su capacidad para procesar y categorizar grandes volúmenes de datos, el modelo permite a los responsables de políticas públicas identificar rápidamente las áreas que requieren atención y ajuste. Este enfoque basado en datos garantiza que las decisiones se tomen en función de necesidades reales, lo que resulta en políticas más efectivas y adaptadas a las problemáticas actuales. Además, la posibilidad de reentrenar el modelo con datos recientes asegura que el sistema se mantenga alineado con los cambios y tendencias emergentes, manteniendo su relevancia a medida que se recopilan nuevas opiniones y reportes de la ciudadanía.

Para la organización encargada de manejar estas problemáticas, contar con esta herramienta significa poder ofrecer un mejor trabajo a la hora de abordar los desafíos globales a nivel local. Con el apoyo de este modelo, la organización puede optimizar sus esfuerzos, logrando una mayor precisión en el seguimiento de las ODS y mejorando su capacidad de respuesta ante temas críticos. En un contexto donde la implementación de políticas públicas debe ser dinámica y adaptativa, esta aplicación no solo facilita el análisis de datos en tiempo real, sino que también permite que las estrategias y planes de acción se ajusten de forma continua y basada en evidencias. Así, el modelo contribuye no solo al fortalecimiento de las políticas públicas en Colombia, sino también a un enfoque de gestión más efectivo y sostenible, alineado con los objetivos de desarrollo global.

Trabajo en Equipo

En el desarrollo de este proyecto, los roles fueron distribuidos de la siguiente manera:

Líder de Proyecto: Daniel Pedroza
Daniel fue responsable de la gestión general del proyecto, definiendo las fechas de reuniones y pre-entregables del grupo. Aseguró que las asignaciones fueran equitativas y verificó el progreso del equipo, además de subir la entrega final. Durante la segunda etapa, dedicó aproximadamente 9 horas a estas tareas. Los principales retos de Daniel fueron coordinar la disponibilidad de los miembros del equipo y gestionar los cambios de último minuto en la documentación. Mantuvo la comunicación constante mediante el servidor de Discord y el grupo de WhatsApp, garantizando que cada integrante cumpliera su parte. Daniel también colaboró en la creación del documento, organizando y consolidando el trabajo final. Además, aportó al desarrollo de la API REST, contribuyendo con aspectos técnicos para asegurar su correcto funcionamiento.

Líder de Analítica: Miguel Gómez
Miguel asumió nuevamente la responsabilidad de gestionar las tareas relacionadas con el análisis de datos y el desarrollo de la API. En esta fase, dedicó cerca de 15 horas a la implementación de la API y a la documentación del proyecto en el archivo README. Los desafíos principales de Miguel fueron simplificar la implementación de la API para cumplir con los requerimientos en el tiempo disponible. Fue una parte fundamental en la parte técnica de la API REST, pero también recibió apoyo de los otros miembros del equipo para lograr un producto final que cumpliera con los objetivos del proyecto.

Líder de Negocios y Datos: Pablo Martínez
Pablo se encargó de garantizar que el enfoque del proyecto continuará alineado con los objetivos del negocio y aportó en la redacción del documento. Dedicó 6 horas a esta tarea, enfocándose en explicar cómo la API puede beneficiar al usuario final y resaltando el valor del modelo analítico. Además, participó activamente en el desarrollo de la API REST, aportando en su funcionalidad y revisando la integración general. Pablo contribuyó a que todos los miembros pudieran colaborar en cada aspecto del proyecto, generando un ambiente de apoyo y aprendizaje mutuo.

Repartición de Puntos
Basado en la contribución de cada miembro del equipo, se decidió repartir los 100 puntos de la siguiente manera:
Daniel Pedroza (Líder de Proyecto): 30 puntos
Miguel Gómez (Líder de Analítica): 40 puntos
Pablo Martínez (Líder de Negocios y Datos): 30 puntos

Reuniones

- **Reunión de planeamiento y preparación**: Definición de roles, objetivos y estrategia para la API.
- **Reunión de revisión técnica**: Evaluación de la implementación de la API y ajustes necesarios.

- **Reunión de documentación**: Asignación de responsabilidades para el README y el documento final.
- **Reunión de revisión de entrega**: Verificación y consolidación del trabajo antes de la entrega del proyecto.

Reflexiones y Mejoras

En el desarrollo del proyecto, seguimos observando la necesidad de reuniones en tiempo real para resolver temas complejos y asegurar una alineación completa entre los miembros. Aunque usamos canales digitales de comunicación, descubrimos que una mayor frecuencia de reuniones presenciales o por videollamada podría mejorar la coordinación y facilitar la toma de decisiones. Para futuros proyectos, se recomienda establecer un equilibrio entre reuniones virtuales y presenciales, lo cual puede reducir malentendidos y optimizar la colaboración del equipo.