Informe

Grupo 23:

- Rafael Santiago Bastos Russi – 202110792
- Jesús Alejandro Davila Pinchao – 202014263
- David Santiago Valderrama Herrera – 201910987

Etapa 1

En el presente documento se presentan algunos apartados que resumen el trabajo realizado. Para mayor detalle, puede consultar la wiki del proyecto.

Entendimiento del negocio y enfoque analítico

Oportunidad/Problema de Negocio	Dentro de los objetivos planteados se encuentra el desarrollo de un modelo analítico que permita discriminar comentarios teniendo en cuenta a cuál objetivo de desarrollo sostenible (ODS) pertenece. Los ODS son objetivos planteados por la Asamblea General de las Naciones Unidas, que fueron diseñados con el fin de abordar muchos de los desafíos globales actuales. Para este proyecto se trabajará sobre los objetivos 3, 4 y 5. El objetivo 3 es sobre la salud y bienestar. En Colombia es un tema complejo que se ha jugado entre las diferentes EPSs privadas. Sin embargo, con el gobierno de Petro se está planteando la posibilidad de que la salud sea administrada por el estado. Esto resulta una oportunidad, ya que conociendo los comentarios que tiene la gente respecto al tercer objetivo se pueden identificar falencias y oportunidades de mejora. El objetivo 4 trata la educación de calidad. En Colombia el acceso a la educación en ciertas zonas del país es complejo, y en aquellas zonas donde puede llegar, no se dicta con la mejor calidad posible. El abordar este objetivo en el país puede traer un gran impacto, pues si una mayor población puede acceder a una educación de calidad, tendrán mejores oportunidades y la calidad de vida mejoraría. El objetivo 5 es la igualdad de género. En Colombia es un tema delicado que ha ocasionado múltiples marchas en inconformidades dentro de las ciudades. Tendría un gran impacto abordar este objetivo, pues de esta manera se reduciría la amplia brecha de género que existe actualmente, y que genera de forma indirecta una discriminación pronunciada a la mujer.
Enfoque Analítico	Con el fin de alcanzar los objetivos propuestos, se concretó el siguiente enfoque analítico: Como lo que se busca es que, a partir de un conjunto de comentarios no organizados con elementos en común (y con una temática acorde con uno de los 3 objetivos previamente mencionados), se puedan generar subgrupos tratando de categorizar cada comentario en uno de los 3 ODS, es necesario aplicar técnicas de aprendizaje automático orientadas a la clasificación. La clasificación permitirá entonces que, a partir de un análisis y una tokenización del texto presente en cada mensaje, se puedan agrupar comentarios similares y asignarles un ODS específico. Los algoritmos que se utilizarán serán: K-Nearest Neighbors (KNN), TF-IDF con Random Forest, y un árbol de decisión.

Organización y rol dentro de ella que se beneficia con la oportunidad definida	La organización es el Fondo de Población de las Naciones Unidas y el rol que se beneficia es el de beneficiario, esta denominación acuña, en el contexto de los objetivos de desarrollo sostenible escogidos a personas vulnerables y desfavorecidas en la dimensión de salud, educación e igualdad de género que han sido integradas a un programa de la organización.
Contacto con experto externo al proyecto	Correo: g.quintana1@uniandes.edu.co Canal: Zoom. Fecha de reunión: Aún sin definir.

Entendimiento y preparación de los datos

Entendimiento

En la distribución de los datos se identificó que los datos contenían caracteres especiales y algunos que carecían de sentido a causa de la codificación en la que se dejaron, además de tener datos numéricos. Se tenían 3000 datos, 1000 por cada Objetivo de desarrollo sostenible (3,4,5), sin filas duplicadas ni valores nulos.

Dada esta distribución identificamos que, para entrenar modelos de clasificación, se debía realizar una preparación de los textos para que puedan ser entendidos por los algoritmos de clasificación.

Preparación

Para un adecuado procesamiento de cada uno de los textos se vio necesario hacer una limpieza de los datos, la cual consistió en poner todos los textos en minúscula y quitar caracteres especiales. Para este proceso se optó por la librería UnicodeData.

Posteriormente se usó la librería spacy para procesar y operar los textos, inicialmente se pretendía usar nltk, pero debido a que está librería no tiene soporte para lenguaje español, fue descartado su uso. Con spacy se creó un objeto Doc que permite operar los textos y con este se eliminaron signos de puntuación y "stop words", por último, se realizó la correspondiente lematización de cada palabra para dejarla en su forma base y facilitar el trabajo a los algoritmos de clasificación.

Así se creó una nueva columna en el dataframe llamada "words" la cual contiene una lista de las palabras del texto completamente procesadas.

Para finalizar se realizó la correspondiente normalización de las palabras para que estas sean entendidas por los algoritmos de clasificación, para esto se usó un vectorizados TF-IDF para tener en cuenta la relación de las palabras con el resto del texto, esto es bastante útil ya que nos da una idea aproximada de la relevancia de la palabra con respecto a las demás presentes en el texto.

Modelado y evaluación

En este apartado se presenta información pertinente con respecto a los algoritmos de clasificación nombrados en el enfoque analítico.

Árbol de decisión (Hecho por Jesús Davila)

Un árbol de decisión es un modelo de machine learning que se utiliza para tomar decisiones basadas en condiciones o reglas. Puede ser representado visualmente como un árbol donde cada nodo

representa una condición o atributo, cada rama representa el resultado de la condición, y cada hoja representa la decisión final o la clase a la que pertenece un dato.

El árbol se construye de manera recursiva dividiendo los datos en cada nodo según las características más significativas. Estas divisiones se realizan de manera que maximicen la pureza de las clases en los nodos hoja.

K-Nearest Neighbors (KNN) (Hecho por Santiago Bastos)

KNN es un algoritmo de clasificación que asigna una etiqueta a un dato basándose en las etiquetas de los "k" vecinos más cercanos en un espacio de características. La idea central es que los datos similares tienden a estar cerca en el espacio de características.

Cuando se presenta un nuevo dato, el algoritmo encuentra los "k" datos más cercanos en el conjunto de entrenamiento y asigna la etiqueta más común o realiza un voto ponderado para determinar la etiqueta del nuevo dato.

TF-IDF con Random Forest (Hecho por David Valderrama)

Este resulta de combinar TF-IDF con Random Forest. Por un lado, TF-IDF es una técnica utilizada comúnmente en procesamiento de texto para asignar pesos a las palabras en una entrada de texto en función de su frecuencia en esa entrada y su rareza en el conjunto de entradas. Por el otro lado, Random Forest es un algoritmo de conjunto que utiliza múltiples árboles de decisión para realizar predicciones. Cada árbol se entrena de manera independiente, y sus resultados se combinan durante la predicción.

El *workflow* sigue como se narra a continuación. Primero, se usa TF-IDF para representar cada entrada de texto como un vector de características, donde cada palabra tiene un peso TF-IDF. Cada entrada de texto se convierte en un vector numérico que puede ser utilizado como entrada para algoritmos de aprendizaje automático. Segundo, una vez obetenida la representación TF-IDF de los documentos, se usa Random Forest como modelo de clasificación. Cada árbol en el bosque se entrena con estos vectores TF-IDF. Finalmente, se obtienen las predicciones y se ha clasificado las entradas de texto.

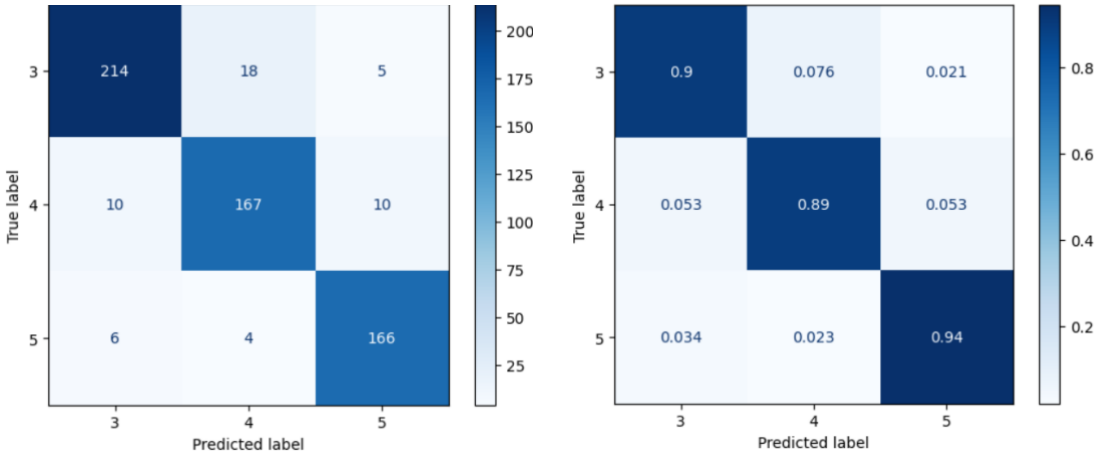
Output de los modelos

Es imperante aclarar que el resultado para todos estos modelos es un pipeline con una clasificación asignada a los textos de prueba suministrados por la organización. Sobre este pipeline se calculan las métricas de calidad, las cuales nos inclinaron a escoger el modelo de TF-IDF con Random Forest. Esta discusión se puede leer en el siguiente apartado del presente documento.

Resultados

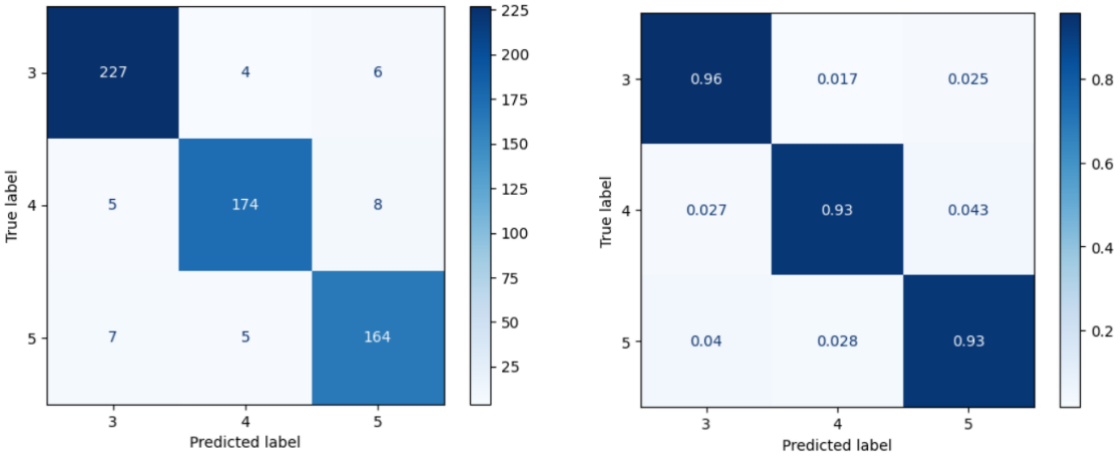
En general, los 3 modelos fueron eficientes y adecuados en la resolución del problema. A continuación, siguiendo el orden respectivo del apartado anterior, se presentan los resultados de cada uno de ellos:

Árbol de decisión



- Teniendo en cuenta las matrices de confusión anteriores, se puede afirmar que es un buen modelo, pues más del 89% de los mensajes fueron categorizados correctamente en cada caso.

K-Nearest Neighbors



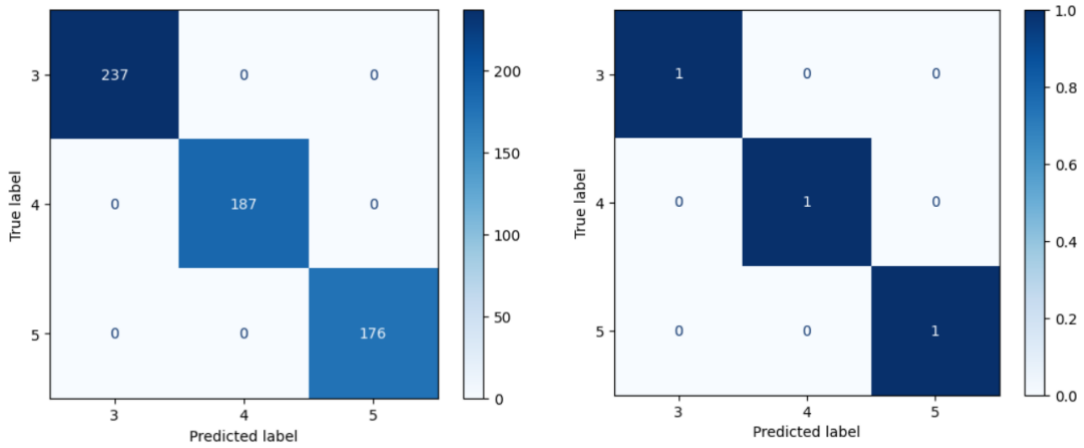
- Teniendo en cuenta las matrices de confusión anteriores, se puede afirmar que es un buen modelo, pues más del 93% de los mensajes fueron categorizados correctamente en cada caso. Además, como se observará a continuación, el modelo tiene una muy buena precisión (del 94%). En términos de recall y f1-score, se tienen métricas favorables, pues no hay un

Precisión: 0.9416666666666667

	precision	recall	f1-score	support
3	0.95	0.96	0.95	237
4	0.95	0.93	0.94	187
5	0.92	0.93	0.93	176
accuracy			0.94	600
macro avg	0.94	0.94	0.94	600
weighted avg	0.94	0.94	0.94	600

overfitting, y la diferencia entre las métricas obtenidas en los datos de prueba y las obtenidas en los datos de entrenamiento no es muy considerable.

TF-IDF con Random Forest



- Teniendo en cuenta las matrices de confusión anteriores, se puede afirmar que aparentemente es un excelente modelo, pues el 100% de los mensajes fueron categorizados correctamente en cada caso. Además, como se observará a continuación, el modelo tiene una muy buena precisión (del 100%). En términos de recall y f1-score, se tienen métricas favorables, pues no hay un overfitting, y la diferencia entre las métricas obtenidas en los datos de prueba y las obtenidas en los datos de entrenamiento es nula.

```
Precisión: 1.0
              precision    recall  f1-score   support

           3       1.00      1.00      1.00       237
           4       1.00      1.00      1.00       187
           5       1.00      1.00      1.00       176

    accuracy                           1.00       600
   macro avg       1.00      1.00      1.00       600
weighted avg       1.00      1.00      1.00       600
```

Mapa de actores

Rol dentro de la empresa	Tipo de actor	Beneficio	Riesgo
Director ejecutivo	Usuario-cliente	Apoya la labor de dirección de toda la organización	Si el modelo no tiene un buen desempeño los esfuerzos de la organización no se organizarán de manera pertinente entre los diferentes

			programas operados por UNFPA.
Líder de programa	Usuario-cliente	Apoya la labor de dirección de un programa operado por UNFPA	Si el modelo no tiene un buen desempeño los programas tendrán problemas con sus objetivos o financiación.
Oficial de colaboraciones	Usuario-cliente	Apoya la labor de solicitud de dinero con base en las estimaciones de un programa de la organización	Si el modelo no tiene un buen desempeño los programas tendrán problemas con su financiación.
Colaboradores	Financiador	Mecanismo de toma de decisiones basado en información	En caso de que el modelo no funcione es dinero mal invertido y pudo dejarse de hacer un proyecto con mayor impacto y viabilidad.
Contratista de tecnología	Proveedor	Garantiza el cumplimiento de estándares de calidad de los productos desarrollados, que incluye elementos como la seguridad y privacidad de los datos utilizados.	Manejo incorrecto de los datos que lleve a la violación de la privacidad de los datos.
Beneficiario	Beneficiado	Recibe ayuda por parte de la organización en materia de salud, educación o igualdad de género.	Afectación personal dado que no se le está suministrado el apoyo correcto y pertinente con respecto a su situación de vulnerabilidad.

Trabajo en equipo

Disponible en la wiki del proyecto.

Etapa 2

Proceso de automatización del proceso de preparación de datos, construcción del modelo, persistencia del modelo y acceso por medio de API

En este apartado se discute el desarrollo de la API y sus especificaciones técnicas.

La API se desarrolló como un servicio en el framework de FastAPI con el fin de poder usar las dependencias de ciencias de datos propias de Python usadas en el desarrollo del modelo de clasificación que se expuso en la anterior etapa del proyecto. Este backend se apalanca del pipeline que contienen el TF-IDF con Random Forest para poder ser consumido por el frontend a fin de ofrecer las funcionalidades de la aplicación al usuario.

Desarrollo de la aplicación y justificación