Proyecto 1: Analítica de Textos

Gabriela Cagua Bolívar - 201812944

Juan Andrés Méndez Galvis - 20181580138

Juan Andrés Romero Colmenares - 202013449

1. Descripción de Roles

Para el desarrollo de este proyecto cada uno de los integrantes asumió un rol específico para que hubiera una buena distribución de trabajo. A continuación, se presentan los roles que desempeñó cada estudiante:

- Juan Andrés Romero:

- Líder de proyecto: Encargado de la gestión del proyecto. Define fechas de reuniones, entregables del grupo, verificación de tareas.
- Líder de analítica: Encargado de la gestión de tareas de analítica
- Horas de trabajo: 60
- Retos afrontados: Tiempo de ejecución de los algoritmos, hubo mucha demora a la hora de correrlos, el GridSearchCV se demoró alrededor de 48 horas para terminar. Para solucionar lo anterior, se decidió disminuir la cantidad de hiperparámetros a variar en el Search, disminuyendo así el tiempo que toma para ser completado

- Juan Andrés Méndez:

- Líder de Datos: Encargado de gestionar los datos a usar y las asignaciones de las tareas sobre ellos
- Horas de trabajo: 24
- Retos afrontados: El entrenar el modelo se demoró 18 horas utilizando GridSearchCV, también había unos caracteres especiales de puntuación que en un inicio el tokenizador no los eliminaba así que toco encontrar la manera de hacerlo. Finalmente, para la gráfica de distribución de palabras se tuvo que entender como traducir la matriz tfidf a un dataset con las frecuencias de cada palabra.

- Gabriela Cagua
  - Líder de negocio: Responsable de velar por resolver el problema o la oportunidad identificada y estar alineado con la estrategia de negocio.
  - Horas de trabajo: 21
  - Retos afrontados: Al principio no fue tan sencillo definir el problema de negocio, sin embargo, se logró solucionar de manera oportuna al solicitar ayuda a un experto en el tema. Debido a que el algoritmo trabajado fue el de regresión logística, no hubo dificultades al respecto

Distribución de puntos entre integrantes del equipo:

De acuerdo con el trabajo realizado por los integrantes del grupo, consideramos que los puntos se distribuirían de acuerdo con el porcentaje de

horas empeñadas para el proyecto, por lo tanto, la distribución sería la siguiente:

Juan Andrés Romero – 57 puntos

Juan Andrés Méndez – 23 puntos

Puntos para mejorar: - 20 puntos

Puntos a mejorar para la segunda entrega:

- Reunirse en momentos más oportunos o mejor planeados
- Destinar más tiempo para el análisis y construcción de los resultados

2. Comprensión del negocio y enfoque analítico:

Oportunidad/Problema de Negocio	El suicidio es una situación de salud pública, siendo la cuarta causa de muerte entre jóvenes de 15 a 29 años[1], que, si es intervenida a tiempo, puede prevenirse mediante ayuda psicológica. Por otro lado, Reddit es una red social que nació en 2005 y permite a usuarios de todo el mundo publicar sus pensamientos de forma anónima, por lo que es un medio para expresar sentimientos y opiniones sobre muchos temas. Teniendo esto en cuenta, miles de personas pueden publicar y han publicado pensamientos o ideas suicidas en Reddit, lo que lo hace un medio ideal para desahogarse por su característica de anonimato, no hay información sobre quien postea, por lo tanto**,** no hay una forma de intervenir. Sin embargo, esta información se encuentra disponible, lo que permite que sea utilizada para detectar si una persona presenta este tipo de ideación y ayudarla, dado el caso.
Enfoque analítico (Descripción del requerimiento del punto de vista de Machine Learning)	Se tiene una base de datos en la que se cuenta con la información de los posts de Reddit de distintos usuarios, relacionados con el tema de la salud mental, más específicamente con la ideación e intento de suicidio, a lo que deben su clasificación. A partir del entendimiento de los datos se pudo observar que se trata de un problema de *analítica de textos,* ya que se tiene que procesar e interpretar el significado de estos para predecir si pertenecen a cierta categoría, razón por la cual también es un problema de clasificación. En este sentido, se aplican distintas técnicas y algoritmos de clasificación de machine learning, para poder predecir si un post es suicida o no.
Organización y rol dentro de ella que se beneficia con la oportunidad definida	La solución de este problema tiene implicaciones relevantes en el mundo real. El suicidio es una situación de salud pública que no solo acaba la vida de la persona que toma la decisión, sino también afecta de manera significativa la de las personas cercanas a esta. En este sentido es importante tomar acciones de prevención - organizaciones enfocadas en ayudar a personas con estos pensamientos para detectar a tiempo estos comportamientos y

[1] Organización Mundial de la Salud, 2021.

	prestar la ayuda psicológica requerida de modo que se puedan salvar las vidas. Organización: Instituciones de salud mental Stakeholders: Usuarios. Los usuarios se pueden beneficiar directamente de este análisis porque se pueden implementar estrategias para brindarles ayuda psicológica en caso de que se detecte que tienen ideación o intento suicida y prevenir esta situación. Psicólogos: esta es una herramienta que puede utilizar un psicólogo para saber si una persona va a cometer suicidio. Por esta razón, a partir de la información de entrenamiento del modelo se puede detectar cuales son las expresiones que indican si la persona va a tomar esta decisión y aplicarlas en pacientes para conocer el estado de la situación. Organización: instituciones educativas Otra aplicación interesante de este análisis son las instituciones educativas como universidades. Es bien conocido que la población que se encuentra en estado de escolarización también es propensa a sufrir de pensamientos e ideación suicida. Stakeholders: 1. Estudiantes. Los estudiantes, se pueden beneficiar de este análisis. Dado a que las universidades prestan servicios de ayuda psicológica, se pueden tomar sus testimonios e interpretarlos de modo que, al realizar la clasificación, se puedan obtener conclusiones sobre el estado de salud mental de esta persona y brindarle la ayuda adecuada.

Algoritmos a utilizar:

Regresión Logística, RandomForestClassifier, MLPClassifier (Red neuronal)

3. Entendimiento y preparación de los datos:

Dentro del tratamiento de datos, para la realización de la tokenización se decidió usar un TweetTokenizer debido a que la naturaleza de los comentarios analizados seguía una estructura bastante similar a la de los posts que se realizan a diario en Twitter. Al principio de la tokenización se decidió excluir a los tokens que pertenecían a los signos de puntuación debido a que eran bastante frecuentes, sin embargo, al comparar las métricas de los modelos sin quitar y quitando los tokens, nos dimos cuenta de que todos los porcentajes bajaban significativamente al removerlos, por lo tanto decidimos dejarlos dentro del vocabulario.

Por otro lado, dado a que casi todos de los mensajes analizados no son ni académicos ni formales, se decidió dejar palabras de “slang”, links y palabras mal escritas ya que no podemos obviar el significado de las frases revisadas.

Una vez realizado esto para comprobar que la tokenización se realizó correctamente, decidimos entrenar más árboles de decisión y una regresión logística para revisar qué tan bien estaban prediciendo y si era necesaria una limpieza más profunda de los tokens, los resultados iniciales fueron de un accuracy de alrededor del 86% y una precisión de 90%. Esto nos mostró que los datos iban en la dirección correcta y que de pronto necesitábamos analizar los hiperparámetros de los algoritmos para mejorar las métricas, por lo tanto, no modificamos más los tokens provistos.

4. Modelado y Evaluación

Para asegurar la calidad y optimización de los modelos revisados, para todos los algoritmos seleccionados se realizó un GridSearchCV de manera que, se pueda encontrar la mejor combinación de hiperparámetros que nos diera los mejores resultados en cuanto a las métricas de error de la clasificación. Para esto se realizó en cada etapa una grilla de hiperparámetros que tuvieran sentido para los distintos algoritmos.

- Regresión Logística – Gabriela Cagua

Explicación del algoritmo: La regresión logística es un algoritmo de machine learning en la que los datos se distribuyen de acuerdo con una distribución binomial. A diferencia de la regresión lineal, cuyos resultados están en una escala numérica y se utilizan las variables predictoras para obtener el valor de las betas, la regresión logística obtiene un resultado binario -Si, No- por lo que se puede concluir que realiza tareas de clasificación, basándose en una función exponencial en oposición a la función lineal utilizada por la regresión lineal. Este tipo de regresión logística se denomina binomial, pero existe regresión logística de tipo multinomial y ordinal.

Para realizar este algoritmo:

1. Se debe seleccionar la variable que se debe predecir, debe ser una variable binaria para que se pueda hacer la clasificación.
2. Se selecciona el conjunto de atributos relevantes para hacer la predicción. Las variables pueden ser numéricas o categóricas. Si son categóricas se hacen tantas variables dummy como categorías, para representar correctamente todos los valores. Al igual que en una regresión lineal, se pueden seleccionar las interacciones. En este caso únicamente se tomaron las variables, no las interacciones.
3. Se entrena el modelo con los datos de prueba. El entrenamiento permitirá al modelo realizar predicciones sobre el conjunto de los datos de prueba.

Después de realizar el GridSearchCV, se encontró que los mejores parámetros para la regresión logística fueron:

C	penalty	solver
10	L2	saga

La regresión con dichos parámetros obtuvo los siguientes resultados

Accuracy	precision	recall	F1-score	support
0.944	0.944	0.957	0.95	39140

De los resultados obtenidos anteriormente, se puede concluir que se encontraron excelentes métricas de error a la hora de entrenar la regresión logística. Debido a que se encontró un nivel de F1Score y accuracy de por encima del 90% (incluso 95% para el F1), se puede decir que el entrenamiento de la regresión fue completamente exitoso. Se recomienda usar este algoritmo debido a que no solo dio muy buenas métricas, sino que también fue el que menos se demoró para encontrar los mejores parámetros.

- Random Forest Classifier – Juan Andrés Romero

Explicación del algoritmo: Los Random Forests son una combinación de un número grande de diferentes árboles de decisión individuales que operan como un conjunto grande para encontrar una buena solución. En esencia, en este algoritmo cada árbol trabaja y es entrenado de manera independiente de manera que cada uno construye su propio camino de decisiones y puede predecir de forma individual. Para seleccionar el resultado de la predicción a retornar, el forest trabaja con un sistema de votación, donde todos los árboles arrojan un resultado y se elige la predicción que haya obtenido más votos.

Para poder encontrar los mejores resultados de este algoritmo, se realizó una búsqueda de tipo GridSearchCV donde los parámetros a buscar fueron el número de estimadores, el criterio de solución y la forma de obtener el máximo número de features a considerar para cada split. Dentro del número de árboles, se trabajaron 10, 100 y 1000 árboles para intentar mirar si hay una tendencia en los resultados, en los max features se trabajó con el modo de raíz cuadrada y logaritmo de 2, y en el criterio se miró Gini y Entropy. Estos dos últimos hiperparámetros fueron seleccionados para revisar si existe alguna variación de las métricas a la hora de resolver y trabajar con los datos en los diferentes árboles de decisión.

En cuanto a los resultados del RandomForest, se encontró que dentro del GridSearchCV los mejores hiperparámetros fueron

N_Estimators	Max_Features	criterion
1000	sqrt	gini

Las métricas obtenidas fueron:

Accuracy	precision	recall	F1-score	support
0.909	0.920	0.919	0.919	39140

Después de evaluar los modelos de RandomForest, se puede ver que se obtuvieron unas métricas bastante aceptables a la hora de revisar las predicciones realizadas. Se puede decir que el entrenamiento del algoritmo fue exitoso debido a que el F1-Score está por encima del 90%. De igual manera, este algoritmo fue el que más se demoró en encontrar los mejores parámetros del GridSearch.

- MLP Classifier – Juan Andrés Méndez

Explicación del algoritmo: Como su nombre indica consiste en una red neuronal de múltiples capas ocultas y neuronas que son definidas por el usuario en conjunto de otros hiperparámetros. El trabajo de una red neuronal es ajustar los pesos y sesgos de cada neurona para que prediga correctamente el valor de salida que les corresponde a los datos de entrada. Para esta arquitectura las neuronas están divididas por capas. Cada neurona recibe el valor de la función de activación multiplicado por el peso de la neurona de todas las neuronas de la capa anterior a esto se le suma el sesgo de la neurona.

El algoritmo consiste en los siguientes principales pasos:

1. Crear una red neuronal con una capa de entrada de la misma forma que las features del dataset. Añadir n capas ocultas cada una con un tamaño de neuronas definido por el analista de datos. Y finalmente hay que asegurar que la capa de salida tenga la misma forma que las variables objetivo. Inicializar los pesos y sesgos de cada una de las neuronas de manera aleatoria. A esto se le llama **Model Phase**
2. Enviar datos de entrada por baches de tamaño b definido por el analista. A esto se le llama el **Forward Pass.**
3. Calcular la función de perdida de los datos que fueron ingresados. Utilizando la función categorical Cross-Entropy. A esta etapa se llama **Calculate Loss.**
4. Después de calcular la perdida propagamos hacia atrás el valor de perdida y ajustamos los nuevos pesos y sesgos basados en la rata de aprendizaje y el tipo de optimizador que se eligió. A esta fase se le llama **Backward Pass**, y es la fase que entrena a la red neuronal.

Teniendo en cuenta cómo funciona el algoritmo se procedió a implementarlo para el data set. Como capa de entrada se tiene las dimensiones de tfidf, la capa de salida es de tamaño 1 ya que solo queremos predecir una variable. Con el fin de encontrar los mejores hiperparámetros para el algoritmo utilizamos la función GridSearchCV que nos permite probar con distintas combinaciones de algoritmos. Los parámetros usados fueron los siguientes.

Fun. activación	alpha	Taza de aprendizaje	Optimizador
Relu, tanh	0.001 , 0.05	Constante, adaptiva	Adam, sgd

Cabe mencionar que se mantuvo el tamaño de capas y neuronas de las capas ocultas constantes de forma **(10, 20, 20).** Los mejores parámetros fueron:

Fun. activación	alpha	Taza de aprendizaje	Optimizador
relu	0.05	constante	Adam

Las métricas obtenidas fueron:

Accuracy	precision	recall	F1-score	support
0.93	0.93	0.93	0.93	39140

Gracias a las métricas obtenidas podemos decir que el entrenamiento del algoritmo fue exitoso ya que el F1-score esta por arriba de 0.9 lo cual es una excelente métrica de clasificación. Se puede evidenciar también que fue el segundo algoritmo más demorado en entrenar. Se recomienda para obtener aún mejores resultados realizar un GridSearchCV variando los tamaños de las capas ocultas.

5. Resultados

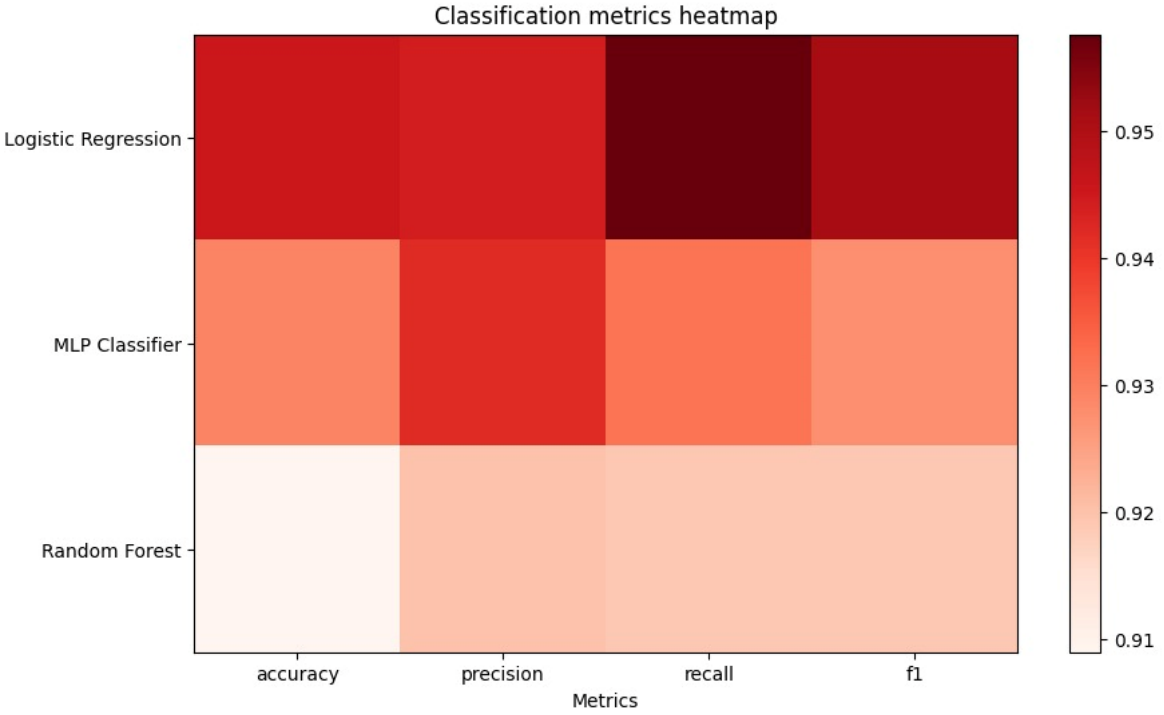
Se puede ver que en general, en todos los algoritmos se obtuvieron muy buenas métricas de error. Lo cual nos indica que cualquiera de los modelos seleccionados nos es útil para resolver el problema de negocio. Por lo tanto, se puede determinar que esta herramienta le permite a cualquier institución educativa o de salud mental intentar predecir la conducta (si es suicida o no suicida), con un alto nivel de precisión de un estudiante o paciente a partir de cualquier mensaje que ellos puedan publicar.

Por otro lado, para la elección del mejor algoritmo se tiene que, si bien todos nos dan excelentes métricas de clasificación hay algunos algoritmos que nos brindan más beneficios que otros. Por ejemplo, RandomForest nos brinda qué tan importantes son ciertos tokens a la hora de clasificar un resultado, mientras que MLPClassfier y Logistic Regression solo nos brindan la respuesta y una predicción del algoritmo, es decir, funcionan como cajas negras y es más difícil sacar información útil de ellos.

Asimismo, con respecto a métricas de error, se tiene que el mejor algoritmo identificado fue el de Logistic Regression con las siguientes métricas de clasificación:

Accuracy	precision	recall	F1-score	support
0.944	0.944	0.957	0.95	39140

Para la identificación de estas métricas se realizó un heatmap conteniendo los resultados de cada modelo, el cual se puede encontrar a continuación. El rojo más intenso señala un porcentaje más alto de la respectiva característica, mientras que un blanco o rojo claro indica un porcentaje de métrica más bajo.



Esto nos muestra que la regresión seleccionada cumple con todos los estándares de métricas y se puede decir que fue entrenado satisfactoriamente.

De igual manera, otra métrica importante para el negocio puede ser el tiempo de entrenamiento y predicción de los algoritmos debido a que es de vital importancia que estos modelos se puedan entrenar en el menor tiempo posible. Esto se debe a que el lenguaje en el que se comunican las personas está en constante evolución y las expresiones que se usaron hace una semana pueden significar algo totalmente opuesto en un corto plazo. De esta manera, se identificó que Logistic Regression también es el modelo más rápido de los 3, demorándose apenas 2 horas en ejecución (utilizando GridSearch) lo cual es más de 10 veces más rápido que el segundo algoritmo más veloz.

Siguiendo con la selección del algoritmo se logró identificar que para cumplir con los objetivos del negocio, realmente no hay ninguna ventaja competitiva de negocio de usar un algoritmo en vez de los otros entrenados. Esto se debe a que, a pesar de que el Random Forest genere información adicional como lo es el feature importance y el árbol de decisión de los tokens, esta información no es lo suficiente significativa para el negocio como para preferir este modelo sobre los otros. Esto se debe a que la información de entrada, es decir, las features o tokens contienen un volumen demasiado alto de información, el cual es aún más complicado de determinar. Para organizaciones como instituciones de salud mental e instituciones educativas, en este caso, la principal prioridad consiste en detectar de manera exitosa una persona que necesite ayuda o que presente algún tipo de comportamiento suicida, para de esta manera brindarle ayuda y poder evitar algún tipo de muerte de este tipo. Este objetivo se puede lograr sin necesidad de comprender a fondo el contenido de cada palabra de manera individual. Por lo tanto, se puede concluir que dado a las métricas y a

lo anteriormente mencionado, se recomienda el uso y aplicación de la Regresión Logística encontrada.

Debido a la naturaleza de los modelos, se le recomienda al negocio utilizarlo como herramienta junto a un análisis de diagnóstico previo, para que de esta manera, se pueda llegar a identificar los comportamientos mencionados anteriormente de forma oportuna, y se pueda actuar con respecto a ellos.

Referencias

Organización Mundial de la Salud (2021). Suicidio. Extraído de https://www.who.int/es/news-room/fact-sheets/detail/suicide