Universidad Nacional de Loja

Faculta de Energía de las Industrias y Recursos Naturales No Renovables

Carrera de Ingeniería en Sistemas

Análisis de Sentimientos en Twitter para la Identificación de Depresión en Tiempos de COVID-19 en Ecuador

Trabajo de Integración Curricular o de Titulación previa a la obtención del Título de Ingeniero en Sistemas

AUTOR:

Byron Stalin Montaño Beltran

DIRECTOR:

Ing. Luis Antonio Chamba Eras, PhD.

Loja – Ecuador

2022

CERTIFICACION

AUTORIA

CARTA DE AUTORIZACION

AGRADECIMIENTO

INDICE DE CONTENIDOS

INDICE DE TABLAS

INDICE DE FIGURAS

1. Título

Análisis de Sentimientos en Twitter para la Identificación de Depresión en Tiempos de COVID-19 en Ecuador

2. Resumen

3. Introducción

La depresión es considerada como la principal causa de discapacidad en todo el mundo y puede afectar a cualquier persona en algún momento de la vida y traer consigo consecuencias devastadoras [1]. Las personas con esta condición suelen utilizar las plataformas de redes sociales como Twitter para expresar sus sentimientos y pensamientos de manera un poco más activa [2], ya que estos sitios de redes sociales permiten que las personas puedan expresar sus opiniones de un tema en específico, sin censura y con mayor libertad. Según la Organización Mundial de la Salud (OMS), aproximadamente 280 millones de personas padecen depresión en todo el mundo [1] [3] , y además ha clasificado a la depresión como el principal factor que contribuye a la discapacidad mundial (más del 7.5% de todos los años vividos con discapacidad en el 2015) y en la sexta posición se encuentran los trastornos de ansiedad [4]. En el Ecuador, durante el año 2020 se realizó un estudio para determinar los efectos de la salud mental y los resultados indican que un número preocupante de personas informó niveles severos o extremadamente severos de depresión, ansiedad y estrés [5]. Además, hay que agregar que debido al brote de COVID-19 que fue declarado una preocupación pública internacional por la OMS el 30 de enero de 2020, seguido por restricciones impuestas por el gobierno al movimiento (encierro), el distanciamiento social y el aumento de casos de muerte amenazó no solo la salud física de las personas, sino también afectó su salud mental, especialmente en términos de emociones [6].
En base a esto, las nuevas tecnologías de inteligencia artificial, como Machine Learning y PLN pueden proporcionar una alternativa de mitigar este problema, ya que ayuda a las computadoras a comprender, interpretar y manipular el lenguaje humano. Y con la cantidad de información existente en las redes sociales como twitter, ésta puede ser analizada mediante el uso de estas tecnologías de una forma mucho más óptima y rápida que cualquier usuario normal podría hacerlo. Una de estas tecnologías es el análisis de sentimientos que hace uso del lenguaje natural y la lingüística computacional para extraer sistemáticamente emociones, sentimientos, opiniones, es decir, la información subjetiva en una pieza de datos textuales [3].

Por lo todo lo expuesto en el párrafo anterior, se planteó el desarrollo de un análisis de sentimientos basado en machine learning que permita identificar si una publicación en twitter tiene contenido depresivo, lo cual permitirá conocer si desde que inició la pandemia de covid-

19 hubo impacto en la salud mental de los usuarios de twitter en Ecuador mediante sus publicaciones.

Bajo este contexto, el Trabajo de Titulación tiene como objetivo principal Identificar publicaciones con contenido depresivo en tiempos de covid-19 en Ecuador mediante el análisis de sentimientos en Twitter, y se encuentra estructurado de la siguiente manera: en el Marco teórico se detallan los conceptos base para adquirir los conocimientos necesarios para el correcto cumplimiento de los objetivos; la siguiente sección es la Metodología en donde se muestra el área de estudio y procedimiento para la realización del trabajo, así como los recursos empleados y los participantes que intervinieron en el desarrollo; en los Resultados se presenta toda la evidencia el desarrollo de los objetivos y siguiendo las fases de la metodología de análisis de sentimientos planteada; en la sección de Discusión se detalla la evaluación de los resultados realizada a partir de la experiencia durante todo del desarrollo de los mismos; finalmente en Conclusiones y Recomendaciones se muestra los resultados relevantes obtenidos y aspectos clave en considerarse en caso de hacer trabajos similares o trabajos futuros derivados del presente Trabajo de Titulación.

4. Marco teórico

En esta sección se presenta la información recolectada que permite una mayor comprensión del contexto que involucra al objeto de estudio. Se inicia en una contextualización general sobre la salud mental y depresión, además se presenta una introducción a la minería de texto y procesamiento de lenguaje natural, además se abarcar algunas generalidades sobre el análisis de sentimientos y sus enfoques principales, también se presentan definiciones sobre algunas herramientas usadas para cumplir el trabajo de titulación y finalmente se muestran algunos trabajos relacionados con el objeto de estudio.

4.1. Salud mental

La salud mental es un componente integral y esencial de la salud. La Constitución de la OMS dice: “La salud es un estado de completo bienestar físico, mental y social, y no solamente la ausencia de afecciones o enfermedades”. Una importante consecuencia de esta definición es que considera la salud mental como algo más que la ausencia de trastornos o discapacidades mentales [7].

4.1.1. COVID 19 y salud mental

El brote de COVID-19 fue declarado una preocupación pública internacional por la OMS el 30 de enero de 2020 [8], que llevo a la mayoría de países a tomar medidas restrictivas para

mitigar la propagación de esta enfermedad. Todas estas restricciones impuestas, como limitación de movimiento (encierro), el distanciamiento social y el aumento de casos de muerte amenazó no solo la salud física de las personas, sino que también afectó su salud mental, especialmente en términos de emociones [9],[6]. La OMS señala que la pandemia ha provocado un incremento de la demanda de servicios de salud mental [9], además otros estudios realizados sobre las emociones con el brote de COVID citaron un aumento de las emociones negativas y el pesimismo entre las personas [7],[6].

4.1.2. Depresión

La depresión es un trastorno de salud mental común que afecta aproximadamente a 280 millones de personas en todo el mundo. Se caracteriza por una tristeza persistente y una falta de interés o placer en actividades que previamente eran gratificantes y placenteras. Además, puede alterar el sueño y el apetito, y es frecuente que concurra con cansancio y falta de concentración. La depresión es una causa importante de discapacidad en todo el mundo, e incide considerablemente en la carga de morbilidad. La depresión puede convertirse en un problema de salud serio, especialmente cuando es recurrente y de intensidad moderada a grave. Puede causar gran sufrimiento a la persona afectada y alterar sus actividades laborales, escolares y familiares. En el peor de los casos, puede llevar al suicidio[1].

4.2. Twitter

Twitter es una popular plataforma de microblogging[2] donde los miembros interactúan entre sí y crean mensajes conocidos como "tweets", posee una base de usuarios grande y en constante crecimiento, por lo que esta plataforma proporciona un rico conjunto de datos en forma de mensajes que generalmente son breves actualizaciones de estado de los usuarios de la aplicación de Twitter que deben expresarse en no más de 280 caracteres de longitud [10]. Debido al límite de longitud en los mensajes publicados en twitter, sus publicaciones son más fáciles de analizar porque los autores suelen ir directos al grano. Por lo tanto, a menudo es más fácil lograr una alta precisión del análisis de sentimientos [11].

4.3. Minería de texto

La minería de textos se define como el proceso de extraer el conocimiento implícito de los datos textuales [12], es decir, describe un conjunto de técnicas lingüísticas, estadísticas y de

[1]Organización Mundial de la Salud. (2021, septiembre 13). *Depresión*. https://www.who.int/es/news-room/fact-sheets/detail/depression

[2] El microblogging es una forma de bloguear a pequeña escala, generalmente compuesta de mensajes breves y concisos. [76]

aprendizaje automático que se refieren generalmente al proceso de extraer información y conocimiento interesante y no trivial a partir de texto no estructurado. Se trata de técnicas que ayudan a modelar y estructurar el contenido informativo de las fuentes textuales para la inteligencia empresarial, el análisis exploratorio de datos, la búsqueda o la investigación [13]. En [14] se afirma que la minería de textos se ha convertido en uno de los campos de moda que se ha incorporado en varias áreas de investigación como la lingüística computacional, la recuperación de información y la minería de datos.

El objetivo principal de la minería de texto es convertir el texto en datos para su análisis, mediante aplicaciones de procesamiento del lenguaje natural y métodos analíticos, y cuando se obtienen estos datos para su análisis, se puede crear un método de procesamiento de estos datos de la manera que se necesite, o ya se han creado bibliotecas para ello [13]. Entre las aplicaciones más significativas que abordan problemas importantes de minería de texto se encuentran el análisis de sentimientos, detección de fraude, filtrado de correo no deseado, entre otros [15].

4.3.1. Minería de texto vs Minería de datos

La minería de texto generalmente se considera como un subdominio de la minería de datos, un campo relacionado con encontrar patrones interesantes en las bases de datos, pero en la configuración del texto. De la misma manera que la minería de datos es parte del proceso global de descubrimiento de conocimiento en bases de datos, la minería de texto podría verse como parte del paradigma de descubrimiento de conocimiento en textos (KDT) [16].

La minería de datos intenta descubrir patrones interesantes a partir de bases de datos masivas. La Minería de texto es el procedimiento de extraer datos e información interesantes y significativos de un texto no estructurado. Es un campo interdisciplinario relativamente nuevo que se interrelaciona con otros campos como la recuperación de datos, la minería de información, el aprendizaje automático, las estadísticas y la lingüística computacional [17].

En definitiva, la minería de texto es mucho más compleja que la minería de datos porque contiene patrones de datos irregulares y no estructurados, mientras que la minería de datos por lo general trata con conjuntos de datos estructurados [14].

4.4. Procesamiento de Lenguaje Natural

El procesamiento del lenguaje natural (PLN) es una rama de la inteligencia artificial que ayuda a las computadoras a comprender, interpretar y manipular el lenguaje humano. La PLN se basa en muchas disciplinas, incluidas la informática y la lingüística computacional, en su búsqueda por llenar el vacío entre la comunicación humana y la comprensión informática [18]. El objetivo principal de la PLN es reconocer, clasificar o extraer la información a nivel

sintáctico y semántico mediante el aprovechamiento de una gran cantidad de computación y datos [19].

4.5. Análisis de sentimientos

El análisis de sentimientos o minería de opiniones se refiere al uso del procesamiento del lenguaje natural y la lingüística computacional para extraer sistemáticamente emociones, sentimientos, opiniones, es decir, la información subjetiva en una pieza de datos textuales. La minería de opiniones ha encontrado su uso principalmente dentro de la investigación de mercado, lo que permite a una empresa comprender el sentimiento con respecto a sus productos y servicios. No solo permite el seguimiento de opiniones sino también de los gustos y disgustos de las personas en general [3]. La tarea del análisis de sentimientos generalmente implica tomar un fragmento de texto, ya sea una oración, un comentario o un documento completo y devolver una "puntuación" que mide qué tan positivo o negativo es el texto [18].

4.5.1. Niveles de análisis de sentimientos

El análisis de sentimiento se clasifica principalmente en tres niveles diferentes [20],[21],[11]:

- **Nivel de documento:** Se trata de etiquetar documentos individuales con su sentimiento. En el nivel de documento, todo el documento se clasifica en clase positiva o negativa.
- **Nivel de oración:** El análisis de sentimiento a nivel de oración trata de etiquetar oraciones individuales con sus respectivas polaridades de sentimiento. La clasificación de sentimiento a nivel de oración clasifica la oración en clase positiva, negativa o neutral. Los textos breves, como el contenido de las redes sociales, se analizan mejor con un análisis de sentimiento a nivel de oración, ya que generalmente consisten en una sola oración.
- **Nivel de aspecto:** Se trata de etiquetar cada palabra con su sentimiento y también identificar la entidad hacia la que se dirige el sentimiento. La clasificación de sentimiento a nivel de aspecto o función se refiere a la identificación y extracción de características del producto de los datos de origen.

4.6. Enfoques para el análisis de sentimientos

Según [20], existen principalmente dos técnicas para el análisis de sentimientos de los datos de Twitter:

4.6.1. Enfoque de aprendizaje automático

El enfoque de aprendizaje automático (machine learning approach) abarca métodos de aprendizaje supervisados y no supervisados. Los métodos supervisados se basan en el uso de conjuntos de datos etiquetados a través de los cuales se crea un modelo para clasificar

los datos de entrada no etiquetados. Los métodos no supervisados se utilizan cuando hay conjuntos de datos sin etiquetar para usar en la fase de entrenamiento, por lo que es necesario emplear algoritmos de agrupamiento para etiquetar los datos [22].

Se han formulado varias técnicas de aprendizaje automático para clasificar los tweets en clases. Las técnicas de aprendizaje automático como Naive Bayes (NB), máxima entropía (ME) y máquinas de vectores de soporte (SVM) han logrado un gran éxito en el análisis de sentimientos [20].

4.6.2. Enfoques basados en léxico

El método basado en léxico (lexicon-based approach) utiliza un diccionario de sentimientos con palabras de opinión y las compara con los datos para determinar la polaridad. Asignan puntajes de sentimiento a las palabras de opinión que describen cuán positivas, negativas y objetivas son las palabras contenidas en el diccionario [20].

Los enfoques basados en el léxico se exploran menos en análisis de sentimientos de Twitter comparado con los métodos de aprendizaje automático. La razón principal es la singularidad del texto en Twitter que no solo contiene una gran cantidad de peculiaridades textuales y expresiones coloquiales, sino que también tiene una naturaleza dinámica con nuevas expresiones y hashtags que surgen de vez en cuando [23].

En [24] se realizó una revisión sobre los enfoques de análisis de sentimiento de Twitter, y determinaron que los trabajos basados en el aprendizaje automático muestran la mayor precisión (98%), entre todos los trabajos estudiados, y los trabajos basados en Deep learning y Léxico se encuentran con un rendimiento satisfactorio, con un máximo de 87,62% y 96,11% respectivamente. Esta comparación se puede ver en la Figura 1 [24].

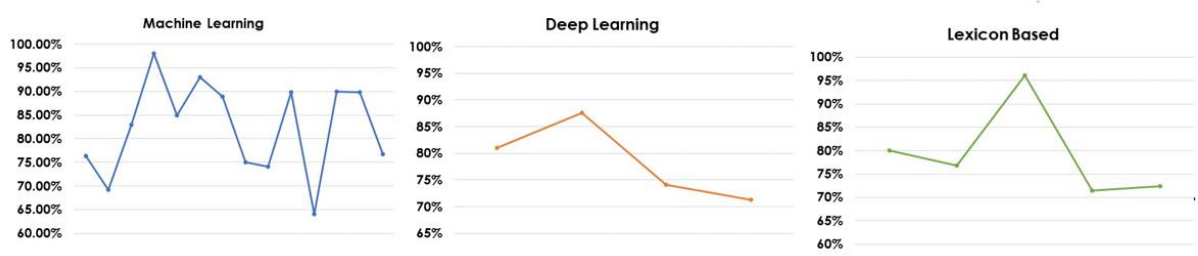


Figura 1. Comparación enfoques de análisis de sentimientos.

4.7. Algoritmos de clasificación

4.7.1. Máquinas de vectores de soporte:

Máquinas de vectores de soporte (Support Vector Machines - SVM) es un algoritmo de clasificación principalmente binaria que se usa típicamente cuando los datos a analizar son limitados. SVM analiza un conjunto de puntos de datos, encuentra un hiperplano que es esencialmente una línea, que puede separar mejor los datos según su tipo o clase. SVM genera varios de estos hiperplanos, pero se elige el que puede separar los puntos de datos de manera óptima, es decir en el que la distancia normal de cualquiera de los puntos de datos es la mayor. Según la posición del hiperplano, los datos se separan en clases separadas [25]. En la Figura 2 se presenta un ejemplo de SVM para dos clases.

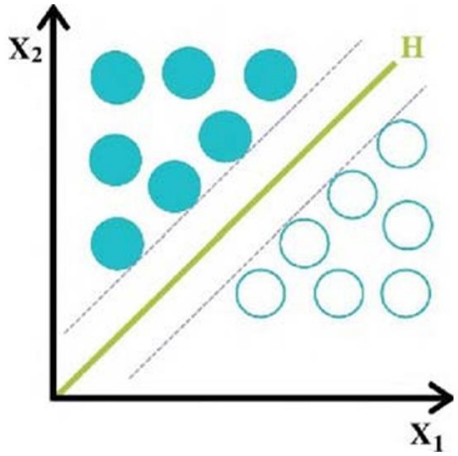


Figura 2. Ejemplo de Maquinas de Vectores de Soporte para dos clases.

La clasificación de texto se adapta perfectamente a las SVM debido a la naturaleza escasa del texto, en el que algunas características son irrelevantes, pero tienden a estar correlacionadas entre sí y, en general, se organizan en categorías linealmente separables [26].

4.7.2. Naive Bayes (NB)

De acuerdo a [27], Naive Bayes es un clasificador probabilístico que utiliza el teorema de Bayes, donde se supone que todas las características (atributos) son independientes entre sí. La siguiente ecuación muestra el modelo NB:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

donde “P(C|X)” es la probabilidad posterior de la clase dada por el predictor, “P(X)” es la probabilidad previa del predictor, “P(C)” es la probabilidad previa de la clase y “P(X|C)” es la probabilidad de la clase de predictor de probabilidad dada.

Aunque el clasificador Naive Bayes es simple, es efectivo debido a su robustez frente a características irrelevantes, además funciona bien en dominios con muchas características importantes, y se considera más confiable para la clasificación de texto y el análisis de sentimientos [28].

4.7.3. Bosque aleatorio

Un clasificador de bosque aleatorio (Random Forest) es esencialmente un conjunto de árboles de decisión. Cada árbol de decisión arroja una "decisión", es decir, una etiqueta que predice la clase de los datos proporcionados. La clase que aparece con mayor frecuencia se elige como etiqueta de los datos. A medida que aumentamos el número de árboles, también aumenta la precisión de la predicción. El clasificador de bosque aleatorio se usa comúnmente para regresión, clasificación y otras tareas. La Figura 3 muestra el funcionamiento básico de un algoritmo de árbol de decisión [25].

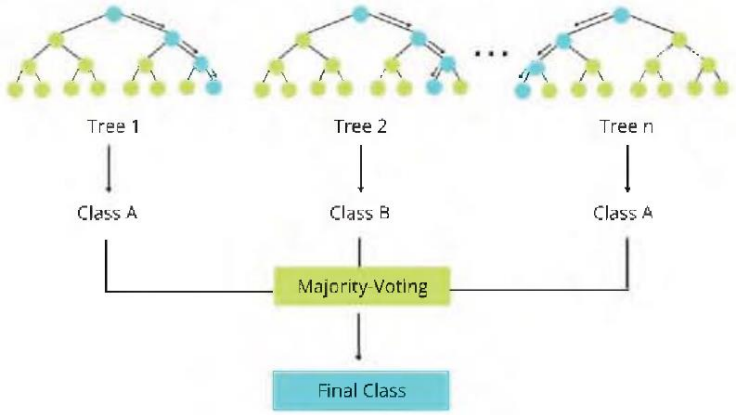


Figura 3. Funcionamiento del algoritmo Bosque aleatorio.

4.8. Python

Python se ha convertido en el lenguaje de programación de facto para el análisis de datos y el aprendizaje automático. Es un lenguaje de secuencias de comandos que se puede usar de forma interactiva y no requiere la compilación del código fuente en un ejecutable para ejecutarse, lo que facilita la transferencia de un programa Python entre computadoras y sistemas operativos [29].

4.9. Jupyter Notebook

Jupyter Notebook (jupyter.org) es una potente herramienta de código abierto basada en navegador para el desarrollo interactivo y la presentación de proyectos de ciencia de datos. Cada cuaderno consta de una colección de celdas ejecutables y cada celda contiene texto formateado usando el lenguaje Markdown o código ejecutable (generalmente Python o R) [30].

4.10. Twint

Twint es una herramienta avanzada de raspado de Twitter (Twitter scraping) escrita en Python que permite raspar Tweets de los perfiles de Twitter sin usar la API de Twitter. Twint utiliza los operadores de búsqueda de Twitter para permitirle recopilar tuits de usuarios específicos, recopilar tuits relacionados con ciertos temas, hashtags y tendencias, u ordenar información

confidencial de tuits como correo electrónico y números de teléfono [31]. Twint también realiza consultas especiales a Twitter, lo que le permite rastrear los seguidores de un usuario de Twitter, los Tweets que le han gustado a un usuario y a quién sigue sin ninguna autenticación o API.

Hay ciertas características de Twint que lo hacen más útil y único de otras API de raspado de Twitter, esto es:

- La API de Twitter tiene restricciones para raspar solo los últimos 3200 Tweets [32]. Twint puede recuperar casi todos los Tweets.
- La configuración es realmente rápida ya que no hay problemas para configurar la API de Twitter.
- Se puede usar de forma anónima sin registrarse en Twitter.
- Es gratis, sin limitaciones de precios.
- Brinda opciones fáciles de usar para almacenar tweets raspados en diferentes formatos: CSV, JSON, SQLite y Elasticsearch.

4.11. Desequilibrio de clases

Al enfrentarse a la situación de crear un modelo de clasificación es habitual que las clases no se encuentran balanceadas, esto puede dar como resultado una precisión bastante alta simplemente prediciendo la clase mayoritaria, pero no logra capturar la clase minoritaria, que suele ser el objetivo de crear el modelo en primer lugar.

Las técnicas de sobremuestreo y submuestreo (figura 4) son muy útiles para resolver la distribución de etiquetas de clase desequilibrada para la clasificación multiclase [33].

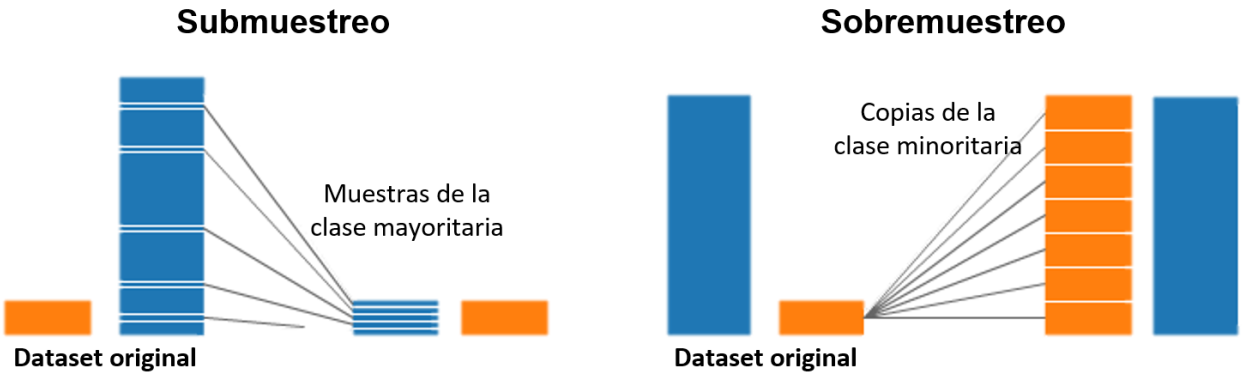


Figura 4. Submuestreo y sobremuestreo en clases desequilibradas.

El **sobremuestreo aleatorio** es un método no heurístico que tiene como objetivo equilibrar la distribución de clases a través de la replicación aleatoria de ejemplos de clases minoritarias.

El **submuestreo aleatorio** también es un método no heurístico que tiene como objetivo equilibrar la distribución de clases mediante la eliminación aleatoria de ejemplos de clases mayoritarias [34].

SMOTE (técnica de sobremuestreo de minorías sintéticas) es otro método popular para realizar el sobremuestreo. En SMOTE, se crean nuevas instancias basadas en la interpolación entre varias instancias de clase minoritaria que se encuentran juntas. Esto permite que SMOTE opere en el espacio de características en lugar de operar en el espacio de datos [35]. Básicamente, la clase minoritaria se puede sobremuestrear creando casos sintéticos en el espacio de características formado por la instancia y sus K vecinos más cercanos, como se muestra en la Figura 5 [36].

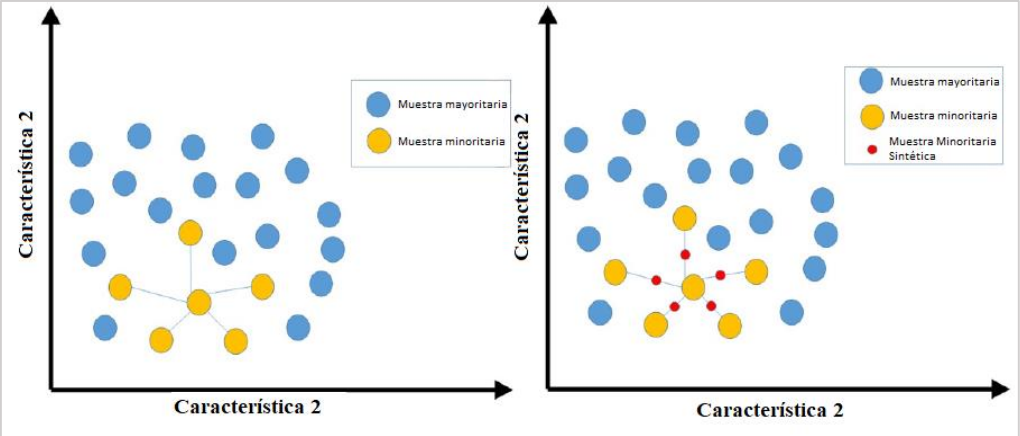


Figura 5. Representación de sobremuestreo usando SMOTE.

El sobremuestreo aleatorio puede aumentar la probabilidad de que se produzca un sobreajuste (overfitting), ya que duplica registros aleatorios de la clase minoritaria. De esta manera un clasificador, por ejemplo, podría construir reglas que aparentemente son precisas, pero que en realidad cubren un ejemplo replicado. Por otro lado, el principal inconveniente del submuestreo aleatorio es que este método puede descartar datos potencialmente importantes de las muestras de la clase mayoritaria [34].

En Python se puede usar la librería “imblearn”, dado que está diseñada específicamente para manejar conjuntos de datos desequilibrados. Proporciona varios métodos como submuestreo (undersampling), sobremuestreo (oversampling) y SMOTE para manejar y eliminar el desequilibrio de un conjunto de datos [35].

4.12. Validación cruzada

La estrategia más típica en el aprendizaje automático es dividir un conjunto de datos en conjuntos de entrenamiento y prueba, el problema con esta estrategia es que no sabemos si una alta precisión de validación indica un buen modelo. Por ejemplo, si realizamos

entrenamiento en el 80% del conjunto de datos y el 20 % restante para fines de prueba, es posible que el 20% de los datos tenga información importante que estamos dejando fuera del entrenamiento en el modelo. Ahí es donde entra la validación cruzada.

- **Validación cruzada de K-Fold**

En este método, se divide los datos de entrenamiento en k subconjuntos (pliegues), se realiza el entrenamiento en todos los subconjuntos dejando un subconjunto (k-1) para la evaluación del modelo entrenado. Luego se va iterando k veces con una división única de entrenamiento:validación [37].

En la figura 6 se puede ver un ejemplo de cómo sería el proceso de validación cruzada de K-fold con 5 iteraciones.

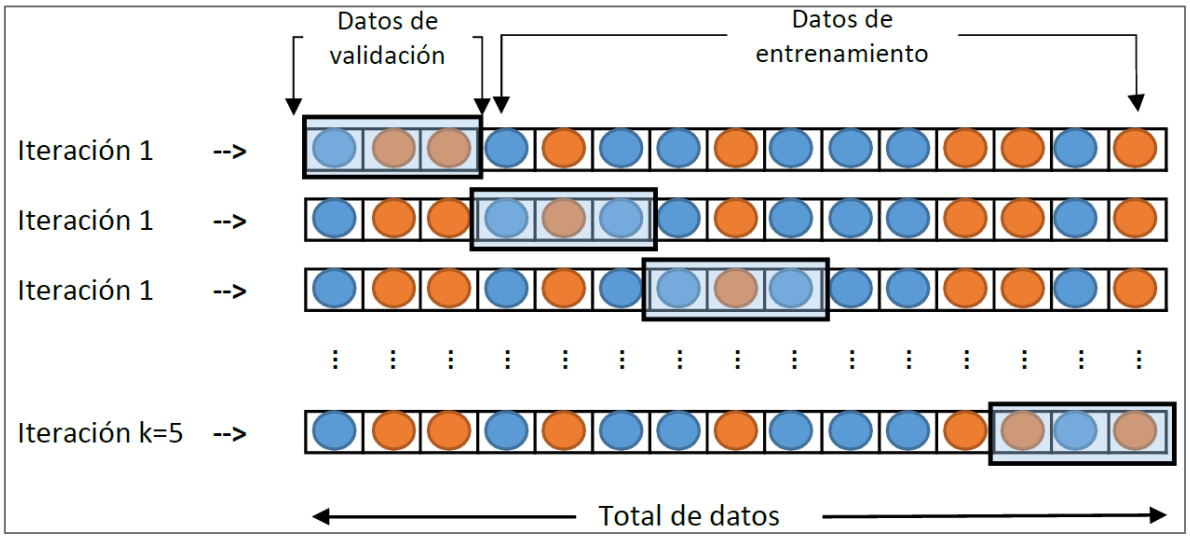


Figura 6. Validación cruzada con 5 iteraciones

En definitiva, la validación cruzada permite garantizar que todas las observaciones del conjunto de datos original tengan la oportunidad de aparecer en la serie de entrenamiento y en la serie de prueba.

4.13. Métricas de evaluación

4.13.1. Matriz de confusión

La matriz de confusión es una medida muy popular utilizada para resolver problemas de clasificación. Se puede aplicar tanto a la clasificación binaria como a problemas de clasificación multiclase [35]. En la Tabla 1 se muestra un ejemplo de una matriz de confusión para la clasificación binaria.

Tabla 1. Ejemplo matriz de confusión

	Clasificado como Positivo	Clasificado como Negativo
Son positivos	**TP**	**FN**
Son negativos	**FP**	**TN**

Las matrices de confusión representan recuentos de valores predichos y reales. La salida "TN" significa True Negative, que muestra el número de ejemplos negativos clasificados con precisión. De manera similar, "TP" significa True Positive, que indica la cantidad de ejemplos positivos clasificados con precisión. El término "FP" muestra el valor de falso positivo, es decir, el número de ejemplos negativos reales clasificados como positivos; y "FN" significa un valor de falso negativo que es el número de ejemplos positivos reales clasificados como negativos [35].

Según [23], las métricas de evaluación utilizadas con más frecuencia son la exactitud, la precisión, la exhaustividad y la puntuación F, adoptadas de los problemas de clasificación tradicionales, estas métricas se muestran a continuación en base a la matriz de confusión (Tabla 1).

- **Exactitud (Accuracy)**: accuracy es la métrica de evaluación más utilizada y mide la frecuencia con la que el método que se evalúa hizo la predicción correcta. Se calcula como la suma de las predicciones verdaderas dividida por el número total de predicciones.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precisión (Precision):** la precisión representa la exactitud del método y se calcula como la proporción de instancias que se pronosticaron como positivas y que lo fueron, dividida por el número total de instancias que se pronosticaron como positivas.

$$Precision = \frac{TP}{TP + FP}$$

- **Exhaustividad (Recall):** que también se conoce como sensibilidad, denota la fracción de instancias positivas que se predijo que serían positivas y se calcula como:

$$Recall = \frac{TP}{TP + FN}$$

- **Puntuación F (F-score):** por lo general, calcular la exhaustividad y la precisión no es suficiente. Una combinación de los dos es más apropiada para evaluar el desempeño de los métodos. La puntuación F es la métrica que combina exhaustividad y precisión.

Esta métrica también se conoce como puntuación F armónica, puntuación F1 o precisión de medida F y se calcula como:

$$F - score = 2 * \frac{precision * recall}{precision + recall}$$

4.14. Trabajos relacionados

En esta sección se encuentran los trabajos relacionados con el objeto de estudio y que fueron obtenidos durante la revisión de literatura.

- En el trabajo llamado *"Identificación de depresión mediante el análisis de sentimientos"* utilizan el análisis de sentimientos para detectar posibles síntomas de depresión por medio de los dispositivos móviles de personas que corran riesgo de sufrir este trastorno mental, para esto obtuvieron sus conversaciones de WhatsApp, que se analizan para detectar posibles emociones y en función del resultado obtenido poder determinar la probabilidad de sufrir depresión [38].
- En "Sentiment Analysis of the COVID-related r/Depression Posts" se presenta un análisis de sentimientos de los mensajes relacionados con COVID publicados en r/Depression en la plataforma social Reddit. Para esto clasificaron los datos de texto sin etiquetar en categorías relacionadas con el contenido del mensaje y dado que las publicaciones no están etiquetadas y es imposible leer una gran cantidad de datos de texto y asignarles un tema relevante de forma manual, utilizaron un modelo estadístico de temas no supervisado (topic model) para asignar un sentimiento relevante a cada publicación. Posteriormente, utilizaron modelos de aprendizaje automático para realizar clasificaciones de texto para examinar y evaluar la exactitud de las etiquetas de los temas y poder realizar un análisis de opinión sobre los datos recién etiquetados [39].
- En el estudio llamado "Depression Detection Using Sentiment Analysis of Tweets" usan análisis de sentimientos para detectar depresión, para lo cual se clasificaron los tweets con rasgos depresivos entre los tweets extraídos de Twitter. Mediante un modelo intentan maximizar la utilización de todas las características lingüísticas disponibles presentes en los tweets y hace uso de técnicas adecuadas de limpieza y preprocesamiento para un diagnóstico más preciso de la depresión. Utilizaron Vader Analyzer basado en reglas y un modelo híbrido de CNN-LSTM [40].
- "An Initiative to Identify Depression using Sentiment Analysis: A Machine Learning Approach", es un proyecto en el cual se propone un algoritmo mediante el cual se

extraen los tweets de Twitter usando R studio y luego se analizan sus sentimientos, es decir, se otorgan puntuaciones a cada sentimiento mediante el cual se identifica si la persona está deprimida o no. Y mediante los conjuntos de datos de Twitter se evalúan a través del algoritmo propuesto. Si el sentimiento ha puntuado como positivo negativo y natural para las emociones positivas, se obtiene una puntuación, pero si la emoción es negativa como ira, disgusto, etc., no obtiene puntuaciones [41].

5. Metodología

5.1. Área de estudio

El presente Trabajo de Titulación (TT) se desarrolló en el cantón Loja, en la Facultad de Energía, las Industrias y los Recursos Naturales no Renovables de la Universidad Nacional de Loja (UNL), en la Carrera de Ingeniería en Sistemas/Computación.

5.2. Procedimiento

Para alcanzar el objetivo general del presente proyecto de titulación se usó el siguiente proceso para cada uno de los objetivos específicos:

1. Construir un conjunto de datos a partir de las publicaciones de Twitter, ver sección 6.1.
    a. Revisión de literatura sobre análisis de sentimientos en Twitter, ver sección 6.1.1.
    b. Definir intervalo de tiempo para la extracción de datos, ver sección 6.1.2.
    c. Recopilación de los tweets mediante la herramienta de scraping Twint, ver sección 6.1.3.
2. Aplicar el análisis de sentimientos mediante una técnica basada en Machine Learning, ver sección 6.2.
    a. Realizar el Preprocesamiento de los datos obtenidos en la fase anterior usando la herramienta de PLN para Python como NLTK, ver sección 6.2.1.
    b. Extracción de características, ver sección 6.2.2.
    c. Detección de sentimiento, ver sección 6.2.3.
3. Interpretar los resultados obtenidos en el análisis de sentimientos, ver sección 6.3.
    a. Evaluar el desempeño de los algoritmos mediante métricas de precisión, accuracy, recall y F1 Score, ver sección 6.3.1.
    b. Predecir contenido depresivo con tweets prepandemia, ver sección 6.3.2.
    c. Realizar un análisis univariado y bivariado de los datos para representar y comparar la cantidad de tuits depresivos, ver sección 6.3.3.

5.3. Recursos

5.3.1. Recursos científicos

Se tomó en cuenta varios métodos y metodologías para la elaboración del presente TT; los cuales se presentan a continuación:

5.3.1.1. Método científico

Es un método de investigación usado principalmente en la producción de conocimiento en las ciencias. Sus principales objetivos son: alcanzar el conocimiento cierto de los fenómenos y poder predecir otros y descubrir la existencia de procesos objetivos y sus conexiones internas y externas para generalizar y profundizar en los conocimientos así adquiridos para demostrarlos con rigor racional y comprobarlos con el experimento y técnicas de su aplicación [42]. Las etapas que integran el método científico son: 1) definición del problema, 2) formulación de hipótesis, 3) recopilación y análisis de datos, 4) confirmación o rechazo de hipótesis, 5) resultados, 6) conclusiones.

Este método se empleó desde la elaboración del marco teórico y durante todo el transcurso de los objetivos del proyecto. En lo que respecta a la definición del problema, en el marco teórico se presenta la relación entre la salud mental y covid-19 y como puede desencadenar en un problema de salud como la depresión. En cuanto a la formulación de hipótesis, esto se hizo acorde a las fases de la metodología KDT, específicamente en la fase Minería de texto y construcción de hipótesis en el desarrollo del objetivo 2, en donde en base a la cantidad de tweets recopilados y preprocesados se planteó la hipótesis que a su vez tiene relación con la pregunta de investigación planteada en este proyecto.

La recopilación y análisis de datos se lo realizo en el objetivo 1 mediante la recopilación de dos clases distintas de tweets, una clase para recopilar tweets que pueden tener indicios de depresión usando palabras clave y que fue depurado con la ayuda de una especialista en el tema de la salud mental, la otra clase de tweets recopilados fueron tweets aleatorios que no contienen características depresivas; en cuanto al objetivo 3 se realizó el análisis de los mejores modelos para identificar depresión y así determinar cuál de ellos ofrece un mejor rendimiento.

Además, para realizar la confirmación o rechazo de la hipótesis planteada, en el objetivo 3 se extrajeron tweets de época prepandemia, específicamente del año 2019 para realizar una clasificación de los tweets depresivos de ese año utilizando el modelo que tuvo mejor rendimiento en el entrenamiento, obteniendo con esto una comparación con los datos ya obtenidos en temporada de covid-19, obteniendo de esta manera los resultados de variabilidad en las publicaciones depresivas en prepandemia (2019) y pandemia (2020 y

2021). Finalmente, las conclusiones se utilizaron para contrastar y presentar los resultados obtenidos en el desarrollo del proyecto.

5.3.1.2. Método analítico

Según [43], el método analítico consiste en distinguir, conocer y clasificar los distintos elementos que conforman un conocimiento general, y a partir del conocimiento general de una realidad realiza la distinción, conocimiento y clasificación de los distintos elementos esenciales que forman parte de ella y de las interrelaciones que sostienen entre sí.

Este método se empleó para conocer las características de los tweets,ya que mediante la recolección de los tweets relacionados a depresión se pudo tener un conocimiento general sobre las publicaciones que se realizan en twitter relacionado al tema, y mediante la investigación se pudo conocer y filtrar las publicaciones que no estaban relacionadas al objeto de estudio para asegurar de que solo hayan tweets que manifiesten depresión mediante la ayuda de un especialista en el área de salud mental, y en base a esto realizar una adecuada clasificación del sentimiento mediante los algoritmos de machine learning utilizados.

5.3.2. Recursos técnicos

5.3.2.1. Entrevista

La entrevista es una técnica que posibilita obtener información acerca de las características de un problema de un informante clave. Dicha información puede ser novedosa o complementaria y ayuda a cuantificar características y la naturaleza del objeto de estudio [44].

Esta técnica se implementó para realizar una entrevista dirigida a la Dra. Ximena Amaya Valarezo, psicóloga clínica, mediante la cual se pudo obtener información relevante que sirvió como sustento y justificación del presente TT, y ya que se realizó a un profesional en cuanto a salud mental se refiere, esta información también contribuyó a resaltar la importancia de los resultados del presente TT. La constancia de la entrevista realizada se puede ver en el anexo 1.

5.3.2.2. Encuesta

Según [45], las encuestas son investigaciones que proporcionan una visión general, mediante la recogida de información estandarizada de una población específica o una muestra representativa de la misma (sujetos del estudio), por medio de un cuestionario o entrevista.
Esta técnica se usó como complemento a la entrevista realizada a la Dra. Ximena Amaya, en este caso se realizó una encuesta dirigida al Dr. Jorge Fernando Jiménez Sánchez, que ostenta el cargo de Psicólogo de la Unidad de Bienestar Universitario en la Universidad Nacional de Loja, mediante esta encuesta se obtuvo información para sustentar y justificar el

presente TT, y dar solidez a la información obtenida en la entrevista mencionada. L evidencia de la encuesta se puede ver en el anexo 2.

5.3.2.3. Metodología KDT

Para el presente TT se tomó como referencia las fases de la metodología KDT, adaptando sus fases a las metodologías propuestas en el análisis de sentimientos, esta metodología se utilizó durante los 3 objetivos para el desarrollo del análisis de sentimientos de acuerdo a cada una de sus fases (sección resultados), las cuales son: Selección de datos, preprocesamiento de datos, transformación de datos, minería de texto y construcción de hipótesis, y finalmente la interpretación/evaluación.

Estas fases se describen a continuación:

Selección de datos.

Esta primera fase se la realizó en el objetivo 1, donde se extrajeron tweets relacionados a depresión y tweets aleatorios de todo el territorio de Ecuador, utilizando palabras clave para recolectar los tweets que pueden ser considerados como depresivos, estos datos fueron el objeto de estudio para las fases restantes de la metodología KDT, el proceso de recolección se realizó de forma automática mediante el uso de la herramienta de scraping Twint.

Preprocesamiento de datos.

Debido a que las publicaciones de Twitter contienen mucho texto informal e incluyen usos idiosincrásicos, es difícil que un algoritmo pueda entender el contexto real de un tweet, es por esto que se realizó una serie de pasos para preprocesar las publicaciones, realizando en primer lugar una limpieza manual para eliminar tweets que no tengan relación al objeto de estudio, para luego eliminar símbolos y caracteres especiales, menciones, conversión a minúsculas, tokenización , eliminación de palabras vacías y lematización. Todo este proceso se realizó al inicio del objetivo 2.

Transformación de datos.

En esta fase se representó todo el texto en números o vectores de números para que puedan ser entrenados con los algoritmos de machine learning, para realizar esto se usó la técnica de tf-idf para convertir el texto en una matriz o vector de características en base a los tweets preprocesados. Todo este proceso se lo realizó para unigramas, bigramas y trigramas, a fin de comparar los resultados con cada uno de ellos en el entrenamiento de los datos.

Minería de texto.

En esta fase los datos se entrenaron mediante los algoritmos Maquinas de Vectores de Soporte, Random Forest y Naive Bayes, cada uno de ellos con unigramas, bigramas y trigramas mediante los vectores de características obtenidos en la fase anterior. El

rendimiento de cada uno de los modelos se validó utilizando la técnica de validación cruzada para asegurarnos que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba. Finalmente se comparó el rendimiento de cada uno de los modelos y se guardó los modelos con mejor rendimiento para usarlos y analizarlos en la fase posterior.

Interpretación/evaluación.

Finalmente, en base a los mejores modelos se generaron graficas de comparación para saber cuál de los modelos ofrece el mejor rendimiento para identificar contenido depresivo, y en base a ese modelo se predijeron publicaciones en tiempo prepandemia, específicamente del año 2019. En base a los resultados de la predicción se compararon con los datos en tiempos de pandemia (2020 y 2021) para determinar la variabilidad en las publicaciones entre los distintos años.

5.4. Participantes

El presente TT enfocado en la línea de investigación de Sistemas inteligentes, se contó con los siguientes participantes.

- Byron Stalin Montaño Beltran, como estudiante investigador y autor del presente TT, iniciando sus actividades desde el planteamiento del tema del PTT, hasta el desarrollo y finalización de los diferentes objetivos establecidos en el presente TT.
- El Ing. Luis Chamba-Eras, Mg. Sc. como director del TT, quien supervisó los avances académicos y técnicos desarrollados por el autor del presente proyecto.
- La Ing. María Del Cisne Ruilova Sánchez, como tutor académico, quien supervisó los avances académicos desarrollados por el autor del presente proyecto.

6. Resultados

En esta sección se detallan los resultados de cada uno de los objetivos específicos del presente TT, obtenidos a través de la aplicación de la metodología planteada en la Fig, en cada uno de los objetivos se detalla las actividades y tareas que se realizaron para dar con el cumplimiento de los mismos.

6.1. Objetivo 1: Construir un conjunto de datos a partir de las publicaciones de Twitter

Para la construcción del conjunto de datos, fue necesario en primer lugar realizar una revisión de literatura sobre el análisis de sentimientos en Twitter, lo cual permitió conocer de una mejor manera la plataforma de recolección de datos, metodología, herramientas, lenguajes de programación y software en general necesario para el análisis de sentimientos. A continuación, se muestra la revisión de literatura y el proceso que se llevó a cabo para obtener los datos de Twitter y construir el conjunto de datos para su posterior uso.

6.1.1. Tarea: Realizar una revisión de literatura sobre análisis de sentimientos en Twitter.

6.1.1.1. Análisis de sentimientos en Twitter

Analizar el sentimiento en Twitter supone asignar a cada mensaje publicado un valor relacionado con la carga emocional que transmite. En relación a esta carga emocional se pueden distinguir algunos tipos de variables [46]:

- **Polaridad**: indica si el mensaje tiene un sentimiento positivo o negativo. En algunos análisis se introduce una tercera categoría para clasificar los mensajes neutros.
- **Intensidad**: proporciona un valor numérico en relación con la intensidad del sentimiento. Se puede distinguir entre una intensidad positiva y una intensidad negativa.
- **Emoción**: clasifica el texto según los distintos tipos de emociones, como puede ser la alegría, la tristeza o la ira.

6.1.1.2. Metodología de análisis de sentimientos

Varios autores plantean metodologías de análisis de sentimientos que varían en 1 o 2 fases. Sin embargo, basado en los estudios [47] [48] [24] [49] [50], una metodología general de análisis de sentimientos en twitter se puede realizar de acuerdo a las siguientes fases:

- Recolección de datos
- Preprocesamiento
- Extracción de características
- Clasificación de sentimientos

- Detección de sentimiento (polaridad)
- Evaluación y análisis

Estas fases se pueden contrastar con una metodología de minería de texto como la metodología KDT que se muestra a continuación.

Descubrimiento de conocimiento en texto (KDT)

Descubrimiento de conocimiento en texto (Knowledge Discovery in Text, KDT) es el proceso que explora grandes conjuntos de datos para identificar patrones útiles y relevantes dentro de ellos. Este proceso también se conoce como minería de datos de texto (Text data mining, TDM) porque puede verse como un proceso de minería de datos que explora datos de texto [15].

El proceso de descubrimiento de conocimiento en texto, implica dominio en diferentes áreas de conocimiento, métodos de recuperación de información, extracción de la información, procesamiento del lenguaje natural y minería de datos. El dominio de estos conocimientos ayuda al investigador a desarrollar cada una de las etapas necesarias previas al descubrimiento de información [51].

De acuerdo a [52] la metodología KDT consta de 5 etapas, estas son:

- **Selección de datos:** consiste en seleccionar los datos adecuados que se procesarán y analizarán en los siguientes pasos.
- **Preprocesamiento de datos:** la tarea es filtrar la información ruidosa y realizar algunos procesos preliminares para facilitar los siguientes pasos, por ejemplo, extraer entidades de nombre de los datos de texto o etiquetar parte del discurso de los datos de texto.
- **Transformación de datos:** los datos de texto se convierten al formato que es fácil de procesar mediante algoritmos de minería, por ejemplo, el formato de vectores, secuencias o tablas de índice invertidas.
- **Minería de texto:** consiste en aplicar algoritmos de minería para encontrar patrones candidatos (palabras o grupos de palabras que se utilizan para identificar conceptos en el texto). Otro autor [53] también considera en esta fase a la construcción de hipótesis.
- **Interpretación/evaluación:** los patrones candidatos producidos en el paso anterior se evalúan y los interesantes se emiten como conocimiento final.

Estas fases se muestran de forma gráfica en la figura 7.

Figura 7. Fases de la metodología KDT

Como se puede ver en la figura 7, el proceso KDT tiene 5 etapas, sin embargo, algunos autores consideran que la metodología KDT se puede agrupar en 3 fases principales, de acuerdo a [54] [15] estas fases son:

- Preparación de texto o datos
- Minería de texto
- Visualización

En consecuencia, los pasos realizados de la metodología pueden variar dependiendo del propósito de la aplicación, no obstante, las fases de análisis de sentimientos propuestas por los autores ya mencionados tienen concordancia con la metodología KDT (Descubrimiento De Conocimiento En Texto), con lo cual se consideró factible aplicar esta metodóloga para el análisis de sentimientos en Twitter. Para tener una mejor perspectiva de la relación entre las fases mencionadas anteriormente se presenta una comparación en la tabla 2.

Tabla 2. Comparación fases de Análisis de sentimientos y KDT.

Fases de análisis de sentimientos	**Etapas de la metodología KDT**
• Recolección de datos	• Selección de datos
• Preprocesamiento	• Preprocesamiento de datos
• Extracción de características	• Transformación de datos
• Clasificación de sentimientos • Detección de sentimiento (polaridad)	• Minería de texto (y construcción de hipótesis)
• Evaluación y análisis	• Interpretación/evaluación

6.1.1.3. Técnicas de extracción de características

En procesamiento del lenguaje natural, cualquier problema basado en texto debe convertirse en una forma que pueda modelarse. El principal problema al trabajar con el procesamiento del lenguaje es que los algoritmos de aprendizaje automático no pueden funcionar directamente en el texto sin formato. Por lo tanto, se necesita alguna técnica de extracción de características para convertir el texto en una matriz (o vector) de características [55].

Algunas de las formas de extracción de características más importantes son:

- Bag-of-Words
- TF-IDF

6.1.1.3.1. Bag of Words (BoW)

Es uno de los métodos más fundamentales para transformar tokens en un conjunto de características. El modelo BoW se usa en la clasificación de documentos, donde la frecuencia de ocurrencia de cada palabra se usa como una característica para entrenar un clasificador [55].

En este modelo, un texto (como una oración o un documento) se representa como la bolsa (bag) de sus palabras, sin tener en cuenta la gramática e incluso el orden de las palabras, pero manteniendo la multiplicidad (número de veces que aparece una palabra) [56].

Hay 3 pasos al crear un modelo BoW:

1. El primer paso es el preprocesamiento de texto.
2. Lo segundo es crear un vocabulario de todas las palabras únicas del corpus.
   Por ejemplo, supongamos que tenemos 2 documentos (d1, d2) que contienen los siguientes datos de texto:
   d1: “Estoy en aprendizaje”
   d2: “Aprendizaje automatico es genial”
   El vocabulario de todas las palabras únicas que se encuentran en estas dos oraciones es:
   v: [estoy, en, aprendizaje, automatico, es, genial]
3. El tercer paso es crear una matriz (o vector) de características asignando una columna separada para cada palabra mientras que cada fila corresponde a un documento.

A continuación, se muestra el ejemplo de la matriz de características del vector v.

Representación en unigramas de d1 y d2:

unigrama(d1) →	**estoy**	**en**	**aprendizaje**	**automatico**	**es**	**genial**
	1	1	1	0	0	0
unigrama(d2) →	**estoy**	**en**	**aprendizaje**	**automatico**	**es**	**genial**
	0	0	1	1	1	1

Cada entrada en la matriz significa la presencia (1) o ausencia (0) de la palabra el texto. Un inconveniente importante al usar este modelo es que se pierde el orden de aparición de las palabras, ya que creamos un vector de tokens en orden aleatorio. Sin embargo, podemos resolver este problema considerando N-gramas (principalmente bigramas) en lugar de palabras individuales (unigramas). Esto puede preservar el orden local de las palabras.

El modelo **N-grama** es un método de comprobación de "n" palabras continuas de una secuencia dada de texto o discurso. Este modelo ayuda a predecir el siguiente elemento de una secuencia. En el análisis del sentimiento, el modelo de n-gramas ayuda a analizar el sentimiento del texto o documento. Unigrama se refiere al n-grama de tamaño 1, bigrama se refiere al n-grama de tamaño 2, trigrama se refiere al n-grama de tamaño 3, y así sucesivamente [57].

Si consideramos todos los bigramas posibles del ejemplo dado, la tabla anterior se vería así:

Y los bigramas de d1 y d2 son:

bigrama(d1) →	**estoy en**	**en aprendizaje**	**aprendizaje automatico**	**automatico es**	**es genial**
	1	1	0	0	0
bigrama(d2) →	**estoy en**	**en aprendizaje**	**aprendizaje automatico**	**automatico es**	**es genial**
	0	0	1	1	1

Sin embargo, esta tabla resultará ser muy grande, ya que puede haber muchos bigramas posibles al considerar todos los pares de palabras consecutivos posibles. Además, el uso de N-gramas puede dar como resultado una matriz dispersa (tiene muchos 0) si el tamaño del vocabulario es grande. Para resolver este tipo de problema se necesita otra técnica, es decir, TF-IDF.

6.1.1.3.2. TF-IDF

Es una técnica de recuperación y extracción de información que tiene como objetivo expresar la importancia de una palabra a un documento que forma parte de una colección de documentos que generalmente llamamos corpus [58].

El TF-IDF se basa en el método de frecuencia y tiene en cuenta la aparición de una palabra en todos los documentos. Penaliza las palabras comunes al asignarles pesos más bajos al tiempo que da importancia a las palabras que aparecen menos en todo el corpus pero que aparecen con frecuencia en pocos documentos individuales. Consta de 2 fórmulas combinadas, la frecuencia de término (TF), que especifica la frecuencia con la que un término aparece en todo el documento y frecuencia inversa de documentos (IDF), que mide si un término es raro o frecuente en los documentos de todo el corpus [59].

Y está formulado como:

$$TF - IDF(t, d) = TF(t, d) * IDF(t)$$

donde “TF(t, d)” es el número de veces que aparece la palabra “t” en el documento “d” y,

$$IDF(t) = log\left(\frac{n}{DF(t)}\right) + 1$$

donde “n” es el número total de documentos, y “DF(t)” es el número de documentos que contienen la palabra "t" [27].

6.1.1.4. Herramientas usadas para el análisis de sentimientos

6.1.1.4.1. Pandas

Pandas es una librería de Python especializada en el manejo y análisis de estructuras de datos de alto nivel diseñadas para hacer que trabajar con datos estructurados o tabulares sea rápido, fácil y expresivo. Desde su aparición en 2010, ha ayudado a que Python sea un entorno de análisis de datos poderoso y productivo. Los objetos principales en pandas que se utilizan son DataFrame, una estructura de datos tabular orientada a columnas con etiquetas de fila y columna, y Series, un objeto de matriz etiquetado unidimensional [60].

6.1.1.4.2. H3

H3 es una solución geoespacial para la partición jerárquica y la indexación espacial en la esfera. Desarrollado por Uber, H3 se ha utilizado activamente como una de las herramientas para las propias necesidades operativas de Uber, que incluye la optimización dinámica de los precios de los viajes y el análisis cuantitativo de los datos geográficos para la toma de decisiones, así como para la visualización. Escrita de forma nativa en C, la librería H3 también tiene una gran selección de enlaces disponibles para otros lenguajes de programación. Estos incluyen, entre otros, C#, JavaScript, Python y R. Una de las características únicas de H3 es la integración de particiones de rejilla de apertura en hexágonos [61].

6.1.1.4.3. NLTK (kit de herramientas de lenguaje natural)

NLTK (Natural Language Toolkit) es una plataforma líder para crear programas de Python para trabajar con datos de lenguaje humano. Proporciona interfaces fáciles de usar para más

de 50 corpus y recursos léxicos como WordNet, junto con un conjunto de bibliotecas de procesamiento de texto para clasificación, tokenización, lematización, etiquetado, análisis y razonamiento semántico, contenedores para bibliotecas NLP de potencia industrial, y un foro de discusión activo [62].

6.1.1.4.4. Stanza

Es una colección de herramientas de procesamiento de lenguaje natural en Python de código abierto que admite 66 idiomas humanos, que introduce una adaptación de su librería CoreNLP a Python.

A pesar de que actualmente existe una considerable variedad de herramientas de PLN disponibles, Stanza fue creada para superar algunas de las limitaciones. En primer lugar, los conjuntos de herramientas existentes suelen ser compatibles con unos pocos idiomas principales. Esto ha limitado significativamente la capacidad de la comunidad para procesar textos multilingües. En segundo lugar, las herramientas de uso generalizado a veces están poco optimizadas en cuanto a precisión, ya sea porque se centran en la eficiencia (por ejemplo, spaCy) o por el uso de modelos menos potentes (por ejemplo, CoreNLP).

Debidos a estos motivos, se seleccionó a Stanza y a NLTK para realizar las tareas de PLN en este proyecto, ya que están basados en Python y sobre todo son multilingüe, con lo cual permite trabajar con texto en español [63].

En la tabla 3, se muestran un resumen de las principales características de algunas bibliotecas comparadas con Stanza y NLTK.

Tabla 3. Bibliotecas de software de PLN y sus características

Librería	**Lenguaje**	**Multilingüe**	**Licencia**
NLTK	Python	Si	Apache 2.0
SpaCy	Python	Si	MIT license
Transformers	Python	Parcialmente	Apache 2.0
CoreNLP	Java	Si	GNU GPLv3
Stanza	Python	Si	Apache 2.0

6.1.1.4.5. Openrefine

Openrefine es una poderosa herramienta para trabajar con datos desordenados: limpiarlos; transformándolo de un formato a otro; y ampliándolo con servicios web y datos externos". Permite la manipulación directa de datos en Wikidata a través de un servicio de reconciliación y una extensión de edición, todo disponible dentro de una interfaz gráfica de usuario y que no

requiere habilidades de codificación. Además, los lanzamientos oficiales de OpenRefine se pueden descargar directamente desde el sitio web de la aplicación [64].

6.1.1.4.6. Librería Emoji

Es una librería de Python que permite imprimir un emoticón en la pantalla usando la función "emojize()" que toma como parámetro el nombre del emoticón encerrado entre dos puntos. En cambio si se desea conocer el texto de un emoticón en particular, se puede usar la función "demojize()" y pasar el emoticón como parámetro [3].

6.1.1.4.7. Scikit-learn

Es una biblioteca de aprendizaje automático de código abierto escrita en Python. Permite la integración fácil y rápida de métodos de aprendizaje automático en código Python. La biblioteca scikit-learn comprende un amplio ancho de banda de métodos para clasificación, regresión, estimación de matriz de covarianza, entre otros [65].

Además, proporciona utilidades para las formas más comunes de extraer características numéricas del contenido de texto. Utilizando la configuración predeterminada de "TfidfTransformer", podemos determinar la frecuencia de termino y la frecuencia inversa de documento (sección 6.1.1.4) [66]. De la misma forma se puede usar el módulo "Tfidfvectorizer" para calcular los valores tf-idf de forma más directa.

En resumen, la principal diferencia entre los dos módulos es la siguiente [67]:

- Con Tfidftransformer, calculará sistemáticamente el conteo de palabras usando CountVectorizer y luego calculará los valores IDF y solo luego calculará las puntuaciones Tf-idf.
- Con Tfidfvectorizer se hace los tres pasos a la vez. De forma interna, calcula el recuento de palabras, los valores IDF y las puntuaciones Tf-idf, todo usando el mismo conjunto de datos.

6.1.1.4.8. Joblib

Joblib es parte del ecosistema SciPy y proporciona utilidades para canalizar trabajos de Python. La API de Joblib2 proporciona utilidades para guardar y cargar objetos de Python que hacen uso de las estructuras de datos NumPy de manera eficiente [68]. En Python, podemos hacer uso de la función joblib para guardar y conservar modelos sklearn. Una vez que el modelo se guarda en el disco o en cualquier otra ubicación, podemos volver a cargarlo o restaurarlo para hacer predicciones sobre nuevos datos [69].

[3] Python, "emoji · PyPI." https://pypi.org/project/emoji/ (accessed Jun. 25, 2022).

6.1.1.4.9. Pickle

El módulo pickle de Python se usa para serializar y deserializar una estructura de objeto de Python. Lo que hace pickle es que primero serializa el objeto antes de escribirlo en el archivo. El decapado (pickling) es una forma de convertir un objeto python (list, array) en un flujo de caracteres. La idea es que este flujo de caracteres contenga toda la información necesaria para reconstruir el objeto en otro script de Python [70].

6.1.2. Tarea: Definir intervalo de tiempo para la extracción de datos.

Desde el surgimiento del covid-19, llegó a tener impacto en el estilo de vida y bienestar de las personas, en Ecuador tuvo mucho impacto sobre todo en el número de muertes provocadas por esta enfermedad. En la figura 7 [4] se puede ver el número de muertes en promedio a nivel de Ecuador desde el año 2020 hasta fines del 2022 en donde se puede notar que las muertes por covid-19 persisten desde el año 2020 hasta fines del año 2021, pero que han ido disminuyendo paulatinamente. Por lo tanto, para la recolección de los tweets se tomó como rango de tiempo desde la fecha en que el covid-19 fue declarado una preocupación pública internacional por la OMS [8], es decir desde el 30 de enero de 2020, y se estableció como fecha límite hasta el 31 de diciembre del 2021, con lo cual se recopilaron publicaciones en Twitter de aproximadamente 2 años.

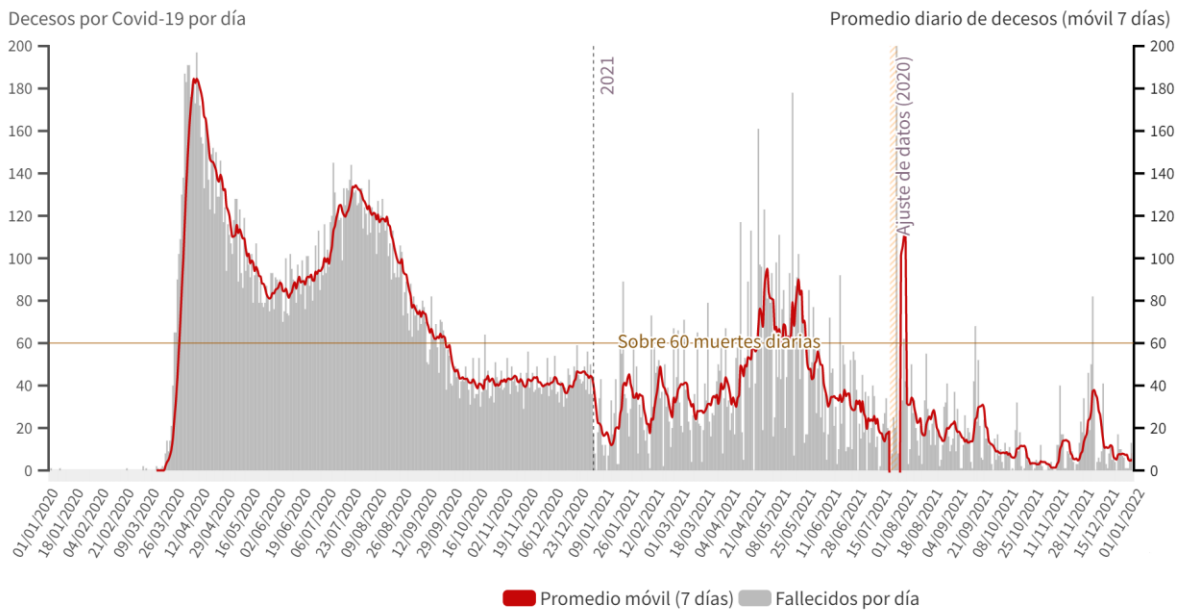


Figura 8. Promedio de muertes por Covid-19 en Ecuador

[4] Observatorio Social del Ecuador, "Personas fallecidas por coronavirus en Ecuador." https://www.covid19ecuador.org/fallecidos

6.1.3. Tarea: Recopilar tweets mediante una herramienta de scraping en Twitter.

Para realizar la recopilación de los tweets, se empieza a ejecutar las fases de la metodología KDT (sección 6.1.1.2), por lo tanto, de aquí en adelante se muestran las fases que se fueron aplicando de acuerdo a las tareas de cada objetivo del presente TT.

Fase 1. Selección de datos

Para la extracción de información en la plataforma de Twitter se utilizó la herramienta de Twitter scraping llamada Twint, ya que brinda libre acceso, no tiene limitaciones en la fecha de extracción de los tweets y no se necesita usar la API de Twitter (ver sección 4.10). Para la extracción de tweets se utilizaron palabras clave para obtener solo la información que es de interés para el proyecto. Para obtener un mejor resultado en la información extraída, y para saber que palabras clave son las adecuadas, se tomó en cuenta lo mencionado en [71], en este estudio identificaron con la ayuda de un psicólogo a las palabras más usadas por personas depresivas en publicaciones de Twitter. También de acuerdo a [72] [73], coinciden en que la depresión es con mayor frecuencia autoinformado y al menos inicialmente autoevaluado, por lo tanto, si una persona dice que está deprimida es muy probable que este deprimida.

En base a los puntos mencionados anteriormente, en la tabla 4 se puede ver las palabras clave que se usó para obtener los tweets que puedan tener indicativos de depresión.

Tabla 4. Palabras clave usadas para la extracción de tweets.

PALABRAS CLAVE
agobiado/a
agotado/a
angustiado/a
ansiedad
decaído
depresión
depresivo/a
deprimido/a
desanimado/a
desesperado/a
desmotivado/a
nervioso
antidepresivo/s
deseperanzado
suicidio

Aparte de la recolección de tweets depresivos, se recolectaron publicaciones aleatorias de twitter que no tengan indicativos de depresión, usando las palabas clave “soy” y “estoy”, ya que representan publicaciones más personales relacionadas al autor del tweet.

Además, para realizar todo el proceso de recopilación de tweets se realizaron las siguientes actividades:

a) Determinar ubicación geográfica de los tweets

El conjunto de datos relacionado con depresión y el conjunto de datos aleatorio se recopiló de usuarios que se encuentran en Ecuador. Para recopilar las publicaciones de las zonas de Ecuador, se utilizó la geolocalización mediante cuadrículas hexagonales de puntos centrales usando la biblioteca H3 de Uber (ver sección 6.1.1.4.2), la cual permitió generar mosaicos hexagonales de aproximadamente el mismo tamaño que cubran el territorio de Ecuador para lograr obtener la mayor cantidad de Tweets disminuyendo el sesgo en la recopilación.

En la figura 9 se puede observar el territorio de Ecuador cubierto mediante los mosaicos hexagonales que se generaron.

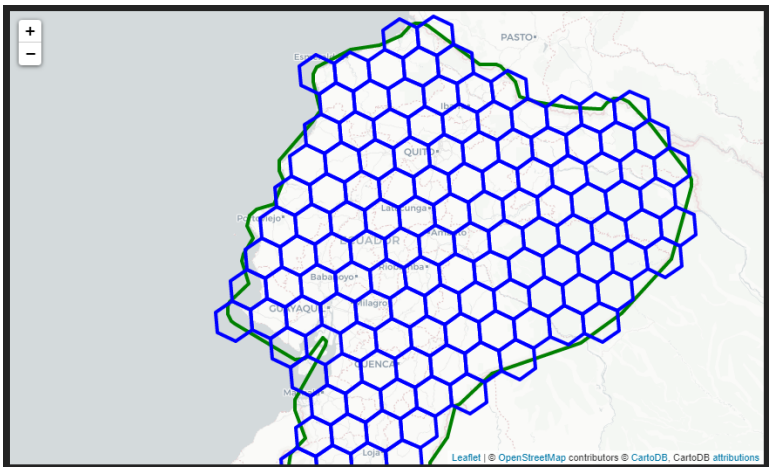


Figura 9. Mapa de mosaicos H3 cubriendo territorio de Ecuador (azul), delimitado por la línea verde.

b) Obtener los Geodatos de Ecuador

Para obtener las coordenadas que se usaron para generar el mapa de mosaicos (figura 9), se usó la herramienta Geojson.io, la cual facilita un entorno interactivo para generar y visualizar datos geográficos (ver anexo 3).

c) Ejecutar el scraping de tweets con la librería Twint

A través de la librería Twint, en donde se usaron como parámetros las palabras clave (ver tabla 4) y las distintas ubicaciones geográficas dentro de Ecuador (ver figura 9), se pudieron obtener un total de 31996 tweets relacionados con depresión. Mediante la recolección de los tweets se formó un dataset que sirvió como fuente principal de datos para el desarrollo de las siguientes etapas.

En cuanto a la recolección de los tweets aleatorios, se utilizó como parámetros una ubicación geográfica central del Ecuador, ya que no se necesitaba tanta precisión en recolectar la mayor

cantidad de datos posibles como es el caso de los tweets depresivos. La obtención se limitó a 10000 tweets aleatorios extraídos. En el anexo 4 se puede observar una previa de los dataset iniciales recopilados.

Cabe recalcar que ambos datasets iniciales tienen datos sin tratar, es decir aun no es factible aplicar algoritmos para un análisis de sentimientos ya que, al ser conjuntos de datos iniciales, contienen mucha información y atributos que son irrelevantes para el objeto de estudio. En la tabla 5 se presentan todos los atributos de cada tuit obtenidos en la recopilación de los dataset iniciales.

Tabla 5. Atributos de los tweets recopilados en el dataset inicial.

Atributos de los tweets recopilados	
id	hashtags
conversation_id	cashtags
created_at	link
date	retweet
time	quote_url
timezone	video
user_id	thumbnail
username	near
name	geo
place	source
tweet	user_rt_id
languaje	user_rt
mentions	retweet_id
urls	reply_to
photos	retweet_date
replies_count	translate
retweets_count	trans_src
likes_count	trans_dest

De todos los atributos de los dataset iniciales mostrados en la tabla 5, la característica “tweet” es la más relevante para el desarrollo del proyecto ya que contienen el texto de la publicación del tweet. Para que estos datos en bruto sean más fáciles de utilizar y aplicar el análisis de sentimiento, se realizó un preprocesamiento de datos que se puede comprobar durante el desarrollo del objetivo 2.

Todo el proceso de recopilación de los tweets desde la generación del mapa de mosaicos hexagonales hasta la ejecución del scraping con Twint se lo realizó en Jupyter notebook, y se puede ver en más detalle en el repositorio [5].

6.2. Objetivo 2: Aplicar el análisis de sentimientos mediante una técnica basada en Machine Learning.

Después de recopilar los datos textuales de Twitter, se desarrolló varias tareas cumpliendo con las siguientes fases de la metodología KDT. En primer lugar, se realizó el preprocesamiento de los datos para eliminar el ruido presente en el texto de los tweets, además se aplicó la extracción de características para convertir el texto en vectores numéricos, y finalmente se realizó el entrenamiento y prueba con los datos usando 3 distintos algoritmos de Machine Learning que son Maquinas de vectores de soporte, Random Forest y Naive Bayes para conocer el comportamiento y rendimiento de cada uno de ellos con los datos existentes.

6.2.1. Tarea: Realizar el Preprocesamiento de los datos obtenidos en la fase anterior.

Fase 2. Preprocesamiento de datos

Las publicaciones de Twitter son diferentes de cualquier otro texto en libros o artículos, ya que incluyen usos idiosincrásicos como menciones de usuarios, retweets, texto informal, entre otros; por lo que se requiere cierto procesamiento a los datos recolectados, por ejemplo en la publicación siguiente: *"#NoMasPresxsPorPlantar #ReglamentosParaTodxs #NoMasPresxsPorFumar #GMM20 #MMMEC20 @Lenin @ottosonnenh Y todo esto hablando del THC, recreacional y espiritual, pero es imposible no mencionar sus cualidades curativas, el CBD es un gran analgésico, Antidepresivo, ansiolítico!! "* es difícil que los algoritmos puedan entender cuál es el contenido real del tweet.

Cabe mencionar que ciertas características para el preprocesamiento ya fueron tomadas en cuenta al momento de la extracción de los tweets usando la herramienta Twint; por ejemplo, no se extrajeron los tweets que contenían links, ya que estos suelen ser usados en su mayoría con fines promocionales, además no se extrajeron los retweets porque por lo general no suelen representar la opinión del autor de esa publicación.

Se siguieron algunos pasos para realizar el preprocesamiento de los datos, en la figura 10 se muestran el flujo que se siguió para realizar el preprocesamiento.

[5] https://github.com/byronmb/Identificacion_Depresion_Ecuador/tree/main/1.Extraccion_Tweets

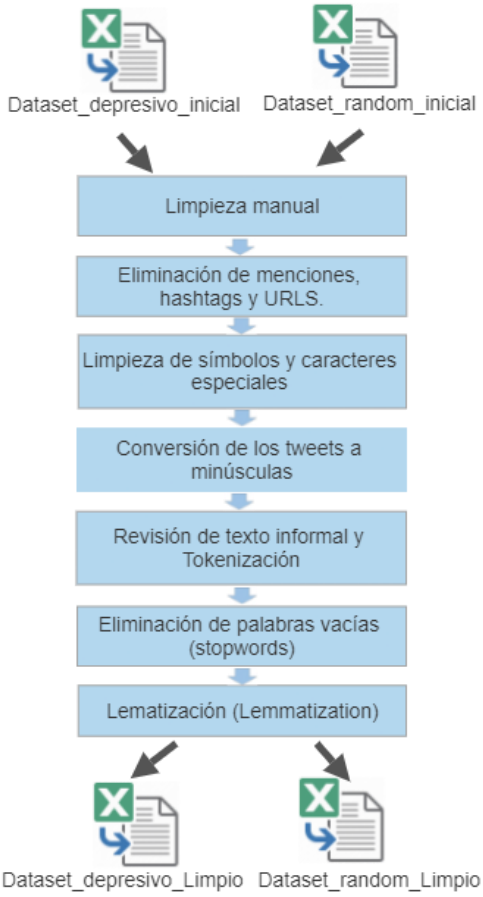


Figura 10. Flujo de trabajo del preprocesamiento de datos

a. Limpieza manual

En primer lugar, se realizó una limpieza manual del dataset relacionado a tweets depresivos ("Dataset_depresivo_inicial") para eliminar tweets que hacen referencia a publicidad, tweets informativos, menciones a otras personas, entre otros. Además, los tweets depurados de forma manual se validaron con la ayuda de la Dra. Sandra Otero, Magister en Psicología Clínica Infantojuvenil[6], para garantizar la calidad de los datos manteniendo solo los tweets que indicaban depresión emocional. En cuanto al dataset de tweets aleatorios ("Dataset_random_inicial"), también se realizó la limpieza manual para eliminar datos que hagan referencia a publicidad, temas políticos, referente a otras personas, entre otros. Aparte de esto, se añadieron 3000 tweets que fueron descartados del dataset depresivo para que el dataset aleatorio contenga palabras relacionadas a depresión, ansiedad, entre otros.

Asimismo, antes de realizar la filtración de las características propias de los tweets, se eliminaron registros duplicados y atributos que no son de utilidad (ver tabla 5).

[6] https://drive.google.com/drive/folders/1WKKCqOgM_PI1aYCbVBTIHoH6-4Zu8vhW?usp=sharing

b. Eliminación de menciones, hashtags y URLS.

Mediante el uso de la sustitución basada en expresiones regulares, se eliminaron los símbolos hashtags (solo símbolo #), y menciones (@nombreusuario) de los tweets. También se procesó las palabras individuales de los tweets; es decir, se eliminaron vocales repetidas presentes en muchas palabras (por ejemplo: “hoolaaaa” a “hola”), tomando en cuenta algunas excepciones como “facebook”, “mood”, “boomerang”, “desee”, entre otras palabras que se descubrieron que estaban presentes en los tweets al realizar la limpieza manual.

Algunos ejemplos de tweets preprocesados con estas características se muestran en la tabla 6.

Tabla 6. Ejemplos de eliminación de menciones, hashtags y urls en los tweets

	Tweet Original	**Tweet después del preprocesamiento**
Eliminación de menciones	**@paolaopioide @CarlosSacoto7 @PoliciaEcuador @teleamazonasec** Solo los que sufrimos de depresion sabemos lo terrible de eso.	Solo los que sufrimos de depresion sabemos lo terrible de eso.
Eliminación del símbolo “#”	Soy fuerte, pero a veces eso de serlo cansa. **#cansado #sad #auxilio #depresion**	Soy fuerte, pero a veces eso de serlo cansa. **cansado sad auxilio depresion**
Eliminación de vocales repetidas	Es bueno escuchar achaque tras achaque, que te señalen **toooodooos** tus errores que te digan recién que todo el ejercicio que estabas haciendo esta en parte mal.? Cuándo te ven casi destruida y deprimida porque hoy viste tus verdaderos resultados en números.	Es bueno escuchar achaque tras achaque, que te señalen **todos** tus errores que te digan recién que todo el ejercicio que estabas haciendo esta en parte mal.? Cuándo te ven casi destruida y deprimida porque hoy viste tus verdaderos resultados en números.

c. Limpieza de símbolos y caracteres especiales.

Los tweets extraídos están llenos de caracteres especiales como “$” y ”-”, que son importantes de eliminar antes de cualquier tipo de análisis de datos, por lo tanto se eliminaron todos los signos de puntuación, símbolos y números que generalmente no suelen contener información que ayude al análisis de sentimientos y que podrían generar resultados impredecibles si se ignoran. También se eliminaron espacios extra presentes en el texto de forma que solo haya un espacio entre palabras, todo esto se realizó mediante el uso de expresiones regulares.

Además, tomando en cuenta que muchos tweets tienen la característica de contener uno o más emoticonos que representan distintos sentimientos de las personas que los escriben, se convirtieron todos los emoticonos a texto en español mediante el uso del paquete de Python “emoji” (Ver sección 6.1.1.4.6).

En la tabla 7, se presentan ejemplos de tweets a los cuales se aplicaron las distintas características mencionadas en este paso del preprocesamiento.

Tabla 7. Ejemplos de limpieza de símbolos, espacios repetidos y conversión de emoticones en los tweets

	Tweet Original	**Tweet después del preprocesamiento**
Eliminación de signos de puntuación y símbolos.	Es normal**,** hay días que entre la depresión**.?..** bueno esta noche es mi caso ☹☹ ganas de gritar**,** salir corriendo y cerrar mis ojos y que ya haya terminado todo este mal	Es normal hay días que entre la depresión bueno esta noche es mi caso ☹☹ ganas de gritar salir corriendo y cerrar mis ojos y que ya haya terminado todo este mal
Eliminación de números	Llevo **13** años con ataques de pánico y ansiedad que suelen darme de manera espontánea y rara vez....es algo que me hace sentir triste por que se que no soy normal como cualquier otro ser humano. En cambio yo vivo con el miedo de cuando me volverá a dar ☹ ataquedepanico	Llevo años con ataques de pánico y ansiedad que suelen darme de manera espontánea y rara vez es algo que me hace sentir triste por que se que no soy normal como cualquier otro ser humano En cambio yo vivo con el miedo de cuando me volverá a dar ☹ ataquedepanico
Eliminación de espacios extra	Estoy desesperada, sin poder gritar ni quejarme con nadie, porque todos estamos igual. Sumidos en desesperación silenciosa. FuckCovid	Estoy desesperada sin poder gritar ni quejarme con nadie porque todos estamos igual Sumidos en desesperación silenciosa FuckCovid
Conversión de emoticones a texto	La depresión es una cosa horrible ☹ y he escuchado hasta a psicólogos tomarlo a la ligera Un abrazo piensa que ahora debe estar mejor	La depresión es una cosa horrible :cara_llorando: y he escuchado hasta a psicólogos tomarlo a la ligera Un abrazo :cara_con_manos_abrazando: piensa que ahora debe estar mejor

En relación a los emoticones presentes en los tweets, se encontró que muchas personas usan varios emoticones repetidos, por este motivo se realizó un preprocesamiento del texto convertido de los emoticones (tabla 7). En primer lugar se eliminó el texto de los emoticones repetidos, también se agregaron espacios entre los emoticones utilizando como puntos de referencia la estructura del texto convertido (:texto_emote:), y por último se eliminaron los dos puntos generados en el texto al momento de la conversión automática realizada en Python.

Los pasos mencionados para depurar los emoticones convertidos a texto se representan mediante ejemplos en la tabla 8.

Tabla 8. Ejemplos de depuración de los textos que fueron convertidos a partir de los emoticones

	Tweet Original	**Tweet después del preprocesamiento**
Eliminar repetidos	Ya me dio una ansiedad maldita **:cara_llorando_fuerte::cara_llorando_fuerte: :cara_llorando_fuerte:**	Ya me dio una ansiedad maldita **:cara_llorando_fuerte:**
Agregar espacios	Y es así como la ansiedad me envuelve otra vez **:cara_decepcionada::cara_desanimada:**	Y es así como la ansiedad me envuelve otra vez **:cara_decepcionada: :cara_desanimada:**
Quitar 2 puntos	No crees en depresión y ansiedad hasta que te pasa **:cara_desanimada:**	No crees en depresión y ansiedad hasta que te pasa **cara_desanimada**

Así mismo, al ser Twitter de naturaleza informal, las personas no acostumbran a hacer uso de las tildes en muchas publicaciones, por esta razón las palabras “relación” y “relacion” son consideradas por los algoritmos como distintas. Para evitar esta pérdida de relación semántica, se eliminaron todas las tildes de las vocales presentes en los tweets y se reemplazaron por su equivalente sin tilde.

En la tabla 9 se presenta un ejemplo de la eliminación de tildes en un tweet.

Tabla 9. Ejemplo de eliminación de tildes en un Tweet.

	Tweet Original	**Tweet después del preprocesamiento**
Eliminación de tildes	Hoy no tengo ganas de nada. Maldita depresi**ó**n	Hoy no tengo ganas de nada Maldita depresi**o**n

d. Conversión de los tweets a minúsculas

Para los algoritmos de aprendizaje no es lo mismo la palabra “depresion” que “DEPRESION”. Estas palabras son tratadas como dos totalmente distintas, sin ningún tipo de relación entre ellas. Para evitar que esto suceda y mantener el significado de las palabras sin tener en cuenta la forma de sus caracteres, todos los tweets se convirtieron a su equivalente en letras minúsculas. En la tabla 10 se muestra un ejemplo de un tweet convertido a minúsculas.

Tabla 10. Ejemplo de conversión de tweet a minúsculas.

	Tweet Original	**Tweet después del preprocesamiento**
Conversión del texto a minúsculas	siento que voy a caer de nuevo en depresión! **AYUDA**	siento que voy a caer de nuevo en depresion **ayuda**

e. Revisión de texto informal y Tokenización

- **Revisión del texto**

Debido al texto informal presente en Twitter, existen muchas palabras que están escritas incorrectamente, ya sea porque están unidas unas con otras, están incompletas o simplemente están mal escritas. Para resolver esto y otros errores que se pasaron por alto en las fases anteriores del preprocesamiento, se exportaron los datasets con el nombre "Dataset_depresivo_semiLimpio" y "Dataset_random_semiLimpio" para procesarlos mediante el uso de la herramienta Openrefine (ver sección 6.1.1.4.5) ya que brinda una interfaz intuitiva para encontrar inconsistencias en las palabras de ambos datasets. Para conocer cómo se realizó este proceso se lo puede ver en el video[7].

- **Tokenización**

Luego de la depuración del texto informal en Openrefine se obtuvieron los dataset "Dataset_depresivo_Openrefine" y "Dataset_random_Openrefine", los cuales fueron utilizados a partir de este paso para continuar con algunas fases más hasta completar el preprocesamiento de los tweets.

En este paso de la Tokenización, se dividió cada tweet en palabras o tokens, es decir se delimitaron las palabras del texto y se convirtieron esas palabras en elementos de una lista para su procesamiento en las siguientes fases. Este proceso se lo realizo mediante el uso de la librería NLTK (ver sección 6.1.1.4.3).

En la tabla 11, se muestra un ejemplo de un tweet que pasó por la fase de Tokenización.

Tabla 11. Ejemplo de un tweet tokenizado.

	Tweet Original	**Tweet después del preprocesamiento**
Tokenización del texto	estoy deprimido todos los dias amanezco con muchas ganas y pienso en que hacer llega el siguiente dia y paso durmiendo todo el dia estoy durmiendo aproximadamente horas al dia parece que estuviera normal pero no cara_sonriendo_ligeramente y no tengo ganas de nada nada nada	['estoy', 'deprimido', 'todos', 'los', 'dias', 'amanezco', 'con', 'muchas', 'ganas', 'y', 'pienso', 'en', 'que', 'hacer', 'llega', 'el', 'siguiente', 'dia', 'y', 'paso', 'durmiendo', 'todo', 'el', 'dia', 'estoy', 'durmiendo', 'aproximadamente', 'horas', 'al', 'dia', 'parece', 'que', 'estuviera', 'normal', 'pero', 'no', 'cara_sonriendo_ligeramente', 'y', 'no', 'tengo', 'ganas', 'de', 'nada', 'nada', 'nada']

f. Eliminación de palabras vacías (stopwords)

Los tweets por lo general contienen muchas palabras que, aunque son necesarias para construir oraciones con sentido, carecen de información que pueden afectar la eficiencia del modelo de manera adversa. Estas palabras son proposiciones, pronombres, conjunciones,

[7] https://youtu.be/M3Lq21P4J74

entre otras; por lo tanto, en esta etapa se aplicó la filtración de palabras vacías o stopwords con el propósito de reducir el número de palabras almacenadas en la lista de tokens procesados en la fase anterior.

La filtración de las palabras vacías se realizó utilizando el corpus de palabras vacías proporcionado por la librería NLTK (ver sección 6.1.1.4.3), excepto algunos pronombres en primera persona y palabras de negación; ya que, dependiendo del tipo de publicación pueden cambiar el significado del texto.

En la tabla 12 se puede observar un ejemplo de filtración de las palabras vacías en un tweet.

Tabla 12. Ejemplo de eliminación de stopwords de un tweet

	Tweet Tokenizado	**Tweet después del preprocesamiento**
Eliminación de stopwords en el texto	['estoy', 'deprimido', 'todos', 'los', 'dias', 'amanezco', 'con', 'muchas', 'ganas', 'y', 'pienso', 'en', 'que', 'hacer', 'llega', 'el', 'siguiente', 'dia', 'y', 'paso', 'durmiendo', 'todo', 'el', 'dia', 'estoy', 'durmiendo', 'aproximadamente', 'horas', 'al', 'dia', 'parece', 'que', 'estuviera', 'normal', 'pero', 'no', 'cara_sonriendo_ligeramente', 'y', 'no', 'tengo', 'ganas', 'de', 'nada', 'nada', 'nada']	['deprimido', 'dias', 'amanezco', 'muchas', 'ganas', 'pienso', 'hacer', 'llega', 'siguiente', 'dia', 'paso', 'durmiendo', 'dia', 'durmiendo', 'aproximadamente', 'horas', 'dia', 'parece', 'normal', 'pero', 'no', 'cara_sonriendo_ligeramente', 'no', 'ganas', 'nada', 'nada', 'nada']

g. Lematización (Lemmatization)

Existen muchas palabras diferentes que representan a una misma palabra. Por ejemplo, las palabras “canto”, “cantas”, “canta” son distintas conjugaciones de un mismo verbo (cantar). Por lo tanto, se aplicó la lematización a las palabras procesadas en la fase anterior para reducirlas a su forma canónica o lema.

Al ser el texto en español, se utilizó la librería Stanza para realizar la lematización de las palabras procesadas porque da mejores resultados para tratar morfología[8] en español (ver sección 6.1.1.4.4). En la tabla 13, se presenta un ejemplo de un tweet preprocesado al que se le aplicó la lematización.

Tabla 13. Ejemplo de texto aplicado la lematización

	Tweet Tokenizado	**Tweet después del preprocesamiento**
Lematización del texto	['a', 'veces', 'es', 'valido', 'sentirse', 'agobiada', 'desesperada', 'y', 'con', 'una', 'incertidumbre', 'tenaz']	['vez', 'valer', 'sentirse', 'agobiada', 'desesperado', 'incertidumbre', 'tenaz']

[8] La morfología estudia la relación entre la forma de las palabras y la información gramatical y semántica que contienen las distintas formas que puede adoptar una misma palabra [77].

Los datasets que fueron procesados mediante todas las fases mencionadas anteriormente se guardaron con el nombre de “Dataset_depresivo_Limpio” para los tweets depresivos y “Dataset_random_Limpio” para los tweets aleatorios.

Finalmente, para fines de visualización de datos, se realizó un análisis de nube de palabras (wordcloud) de los tweets preprocesados en ambos conjuntos de datos.

Figura 11. Nube de palabras de los tweets alestorios.

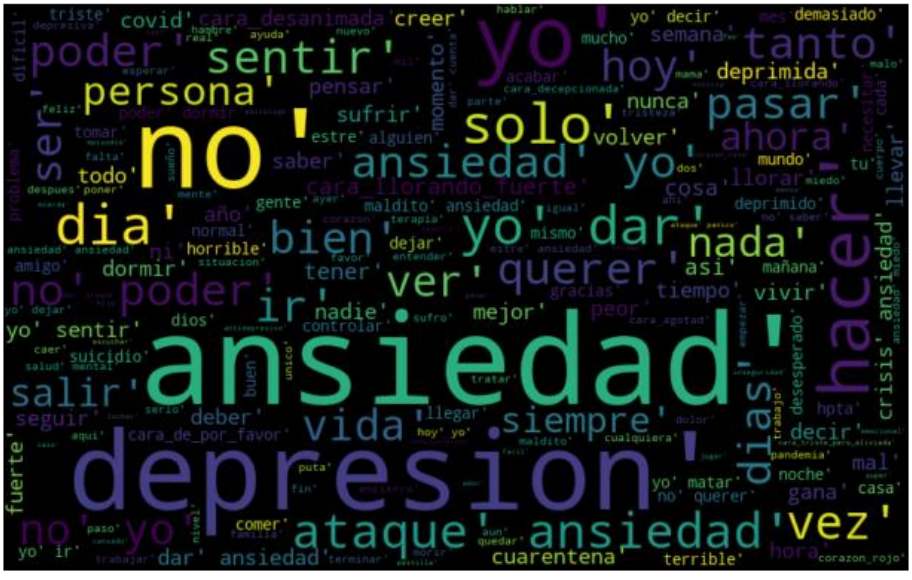


Figura 12. Nube de palabras de los tweets depresivos

La figura 11 muestra la presencia de palabras encontradas en el conjunto de datos formado a partir de tweets aleatorios que no contienen ningún rasgo que indique depresión y se puede observar diversas la variedad de palabras que están presentes con más frecuencia, se puede

ver palabras como ansiedad y depresión que resaltan debido a la inclusión de tweets descartados del dataset depresivos que contienen estas palabras pero que no expresan tendencias depresivas. La figura 12 en cambio muestra claramente la presencia de ciertas palabras que representan una emoción negativa y la presencia de tales palabras en el lenguaje de una persona es un indicador de tendencia depresiva.

Todo el proceso de preprocesamiento a los datos recolectados fueron realizados en Openrefine y en Jupyter Notebook, este proceso en detalle se puede ver en el repositorio[9].

6.2.2. Tarea: Extracción de características

Fase 3. Transformación de datos

Los algoritmos de aprendizaje automático funcionan con valores numéricos, por lo tanto, primero se tiene que representar el texto en números o vectores de números. Para realizar esto se usó la técnica de tf-idf (ver sección 6.1.1.3) para convertir el texto en una matriz o vector de características en base a los tweets preprocesados.

En primer lugar, se cargaron los datasets (“Dataset_depresivo_Limpio”, “Dataset_random_Limpio”) que fueron preprocesados en la sección anterior (sección 6.2.1), y se etiquetaron de acuerdo a la característica de los tweets, estas son:

- Clase 0 para tweets con sentimiento aleatorio (no depresivo)
- Clase 1 para tweets con sentimiento depresivo.

Luego se unificaron y se mezclaron ambos datasets clasificados para generar un solo conjunto de datos que contiene tanto tweets depresivos como tweets no depresivos. El conjunto de datos unido y etiquetado se almacenó en un archivo llamado “Dataset_unido” para uso posterior, quedando un total de 10750 tweets en el conjunto de datos, en la figura 13 se muestra una representación de la cantidad de tweets clasificados con su respectiva clase.

[9] https://github.com/byronmb/Identificacion_Depresion_Ecuador/tree/main/2.Preprocesamiento

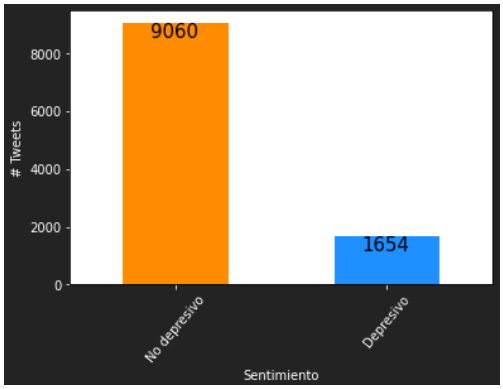


Figura 13. Tweets clasificados como aleatorios y depresivos.

Como se puede ver en la figura 13, los datos que fueron recolectados y preprocesados muestran un desequilibrio en sus clases; ya que, en la limpieza de los tweets depresivos recolectados, la cantidad de datos se redujeron significativamente; por lo tanto, para garantizar que los algoritmos de aprendizaje no estén sesgados hacia la clase mayoritaria, se aplicó la técnica de sobremuestreo de minorías sintéticas SMOTE (ver sección 4.11) utilizando “over_sampling” de la librería imblearn para equilibrar la distribución de las clases en el conjunto de datos, como se puede ver en la figura 14.

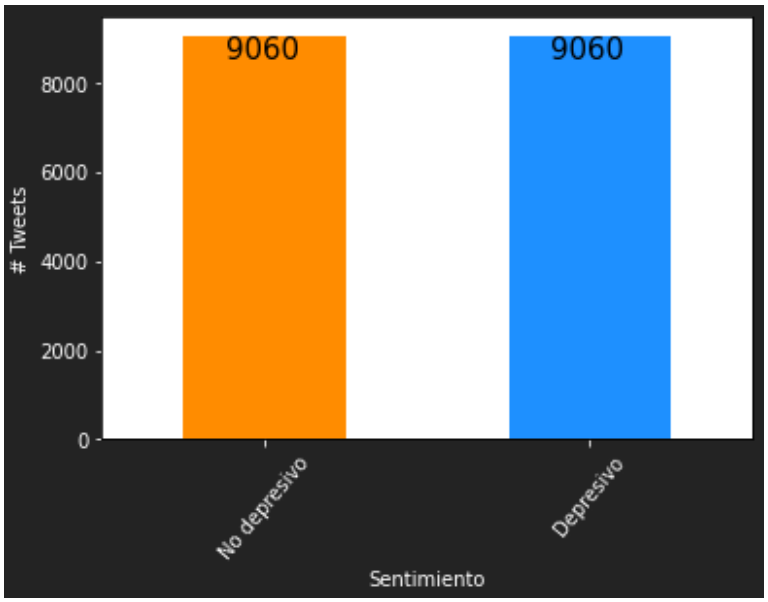


Figura 14. Conjunto de datos con las clases equilibradas

Se realizó la extracción de características utilizando “TfidfTransformer” de la librería sklearn (sección 6.1.1.4.7) para conocer de forma más clara y detallada este proceso. Se realizó el cálculo de la frecuencia de términos (TF), la frecuencia inversa de documentos (IDF) para finalmente obtener los valores tf-idf del conjunto de datos. Todo este proceso para obtener el TF-IDF vectorizado se lo realizó para unigramas, bigramas y trigramas, a fin de comparar los

resultados con cada uno de ellos en la fase de entrenamiento. Los pasos en detalle se muestran a continuación:

6.2.2.1. TF-IDF para unigramas

- **Calculo de la frecuencia de términos (TF)**

Mediante el módulo CountVectorizer se calculó el número de veces que aparece cada palabra en cada uno de los documentos (tweets), y se guardó el vocabulario de todos los unigramas posibles con el nombre de "vocabulary_Unigrama.pkl" para uso posterior en posibles predicciones con nuevos datos, esto se realizó mediante el uso de la librería Pickle (ver sección 6.1.1.4.9). En la tabla 14, se muestra una previa de la tabla generada del recuento de todas palabras del conjunto de datos.

Tabla 14. Recuento de palabras (TF) del conjunto de datos.

	ab	**abajo**	**abandonar**	**abandono**	**….**	**zurda**	**zurdo**	**ñaña**	**ñaño**
0	0	0	0	0	….	0	0	0	0
1	0	0	0	0	….	0	0	0	0
2	0	0	0	0	….	0	0	0	0
3	0	0	0	0	….	0	0	0	0
4	0	0	0	0	….	0	0	0	0
….	….	….	….	….	….	….	….	….	….
10709	0	0	0	0	….	0	0	0	0
10710	0	0	0	0	….	0	0	0	0
10711	0	0	0	0	….	0	0	0	0
10712	0	0	0	0	….	0	0	0	0
10713	0	0	0	0	….	0	0	0	0

El recuento de palabras (unigramas) obtuvo como resultado 10714 filas, que es el total de documentos o tweets y 7152 columnas, es decir el total de palabras únicas (unigramas) que están presentes en todo el dataset y que están presentes en al menos 2 documentos (min_df=2).

Como se puede ver en la tabla 14, la mayoría de los elementos en el vector de características son 0, por lo que se genera una matriz dispersa. Esto se debe a que el número de palabras que aparecen en cada documento (tweet) es solo una pequeña parte de todo el conjunto de documentos; por lo tanto, hay muchas palabras que no aparecen y se marcarán como 0.

- **Calculo de los valores IDF**

Se calculó los valores IDF en base a los recuentos de palabras que se obtuvieron anteriormente. En la tabla 15 se muestra algunos valores IDF que se obtuvieron mediante un marco de datos de Python, estos valores se presentan ordenados de forma ascendente.

Tabla 15. Previa de los valores idf para unigramas del conjunto de datos

	valores_idf
no	1.989609
yo	2.209745
ansiedad	2.612710
hacer	3.196851
depresión	3.330503
...	...
piloto	9.180788
Pilsener	9.180788
Pincel	9.180788
Escobo	9.180788
Descontrolado	9.180788

Como se puede observar en la tabla 15, las palabras “no” y “yo” tienen los valores IDF más bajos. Esto quiere decir que estas palabras aparecen en la mayoría de los documentos de la colección, ya que cuanto menor sea el valor IDF de una palabra, menos única será para cualquier documento en particular.

Además, hay que tener en cuenta que la formula TF-IDF de la notación estándar de los libros de texto (ver sección 6.1.1.3.2) es diferente a la fórmula de cálculo de sklearn, en donde se agrega la constante “1” al numerador y denominador lo cual evita las divisiones por cero. Por lo tanto, la fórmula queda de la siguiente manera:

$$idf(t,d) = \log \frac{n_d + 1}{1 + df(t,d)}$$

$$tf - idf(t,d) = tf(t,d) * (idf(t,d) + 1)$$

- **Calculo de las puntuaciones TF-IDF**

Una vez que se calculó los valores IDF, ya se puede calcular las puntuaciones tf-idf para cualquier documento (tweet) o conjunto de documentos.

Se realizó el cálculo de las puntuaciones tf-idf para todos los documentos del conjunto de datos. En la tabla 16 se presenta una previa de la representación de valores tf-idf de un documento (tweet) que se colocó en un marco de datos para fines de visualización y se ordenaron sus puntajes en orden descendente.

Tabla 16. Valores tf-idf de un tweet del conjunto de datos

	tfidf
gym	0.597944
ayudar	0.490538
matar	0.458744
deber	0.381294
ansiedad	0.214487
...	...
divertido	0.000000
diversión	0.000000
distraído	0.000000
distraída	0.000000
ñaño	0.000000

En la tabla 16, se puede notar que solo ciertas palabras tienen puntaje tf-idf, ya que son solo las que aparecen en este documento en particular. Todas las palabras de este documento tienen una puntuación tf-idf y todo lo demás aparece como ceros, ya que cuanto más común sea la palabra en todos los documentos, menor será su puntuación; por ejemplo, en este caso la palabra “ansiedad” tiene una puntuación menor en comparación a las demás palabras, esto quiere decir que esta palabra está presente en muchos más documentos. Además, cuanto más exclusiva sea una palabra en el actual documento (por ejemplo, "gym" y "ayudar"), mayor será la puntuación.

Se obtuvo las variables independientes y dependientes aplicando el sobremuestreo con SMOTE, para la variable independiente se usó el vector de características tf-idf obtenido y para la variable dependiente se usó la etiqueta sentimiento (0 para no depresivo y 1 para depresivo).

Finalmente, las variables dependientes e independiente obtenidas se almacenaron con el nombre de “x_tfidf_Unigrama” y “y_tfidf_Unigrama” respectivamente para uso posterior en el entrenamiento y prueba de los modelos.

6.2.2.2. TF-IDF para bigramas

- **Calculo de la frecuencia de términos (TF)**

Mediante el módulo CountVectorizer se calculó el número de veces que aparece cada bigrama (2 palabras consecutivas) en cada uno de los documentos (tweets), y se guardó el vocabulario de todos los bigramas posibles con el nombre de “vocabulary_Bigrama.pkl” para uso posterior en posibles predicciones con nuevos datos, esto se realizó mediante el uso de la librería Pickle (ver sección 6.1.1.4.9). En la tabla 17 se muestra una previa de la tabla generada del recuento de todos los bigramas del conjunto de datos.

Tabla 17. Recuento de bigramas (TF) del conjunto de datos.

	abajo no	abandonar serie	abogado contraparte	aborto asesinato	...	yo volvi	yo yo	you can	zapatitos rojos
0	0	0	0	0	...	0	0	0	0
1	0	0	0	0	...	0	0	0	0
2	0	0	0	0	...	0	0	0	0
3	0	0	0	0	...	0	0	0	0
4	0	0	0	0	...	0	0	0	0
...	...	...	...	...	...	...	...	...	...
10709	0	0	0	0	...	0	0	0	0
10710	0	0	0	0	...	0	0	0	0
10711	0	0	0	0	...	0	0	0	0
10712	0	0	0	0	...	0	0	0	0
10713	0	0	0	0	...	0	0	0	0

El recuento de bigramas obtuvo como resultado 10714 filas, que es el total de documentos o tweets y 9043 columnas, es decir el total de bigramas únicos que están presentes en todo el dataset y que están presentes en al menos 2 documentos (min_df=2).

Como se puede ver en la tabla 17, al igual que con los unigramas, la mayoría de los elementos en el vector de características son 0, por lo que se genera una matriz dispersa. Esto se debe a que el número de bigramas que aparecen en cada documento (tweet) es solo una pequeña parte de todo el conjunto de documentos; por lo tanto, hay muchos bigramas que no aparecen y se marcaran como 0.

- **Cálculo de los valores IDF**

Se calculó los valores IDF en base a los bigramas que se obtuvieron anteriormente. En la tabla 18 se muestra algunos valores IDF que se obtuvieron mediante un marco de datos de Python, estos valores se presentan ordenados de forma ascendente.

Tabla 18. Previa de los valores idf para bigramas del conjunto de datos

	valores_idf
no yo	4.194900
yo dar	4.323563
no poder	4.468259
yo decir	5.069914
ansiedad yo	5.337757
...	...
hecho bolita	9.180788
harto mascarilla	9.180788
harto leer	9.180788
hombre parecer	9.180788
zapatitos rojos	9.180788

Como se puede observar en la tabla 18, los bigramas “no yo” y “yo dar” tienen los valores IDF más bajos. Esto quiere decir que estos bigramas aparecen en la mayoría de los documentos de la colección, ya que cuanto menor sea el valor IDF del bigrama, menos única será para cualquier documento en particular.

- **Cálculo de las puntuaciones TF-IDF**

Una vez que se calculó los valores IDF para bigramas, ya se puede calcular las puntuaciones tf-idf para cualquier documento (tweet) o conjunto de documentos.

Se realizó el cálculo de las puntuaciones tf-idf para todos los documentos del conjunto de datos. En la tabla 19 se presenta una previa de la representación de valores tf-idf para bigramas de un documento (tweet) que se colocó en un marco de datos para fines de visualización y se ordenaron sus puntajes en orden descendente.

Tabla 19. Valores tf-idf (bigramas) de un tweet del conjunto de datos

	tfidf
hacer ejercicio	0.596923
ejercicio deber	0.384000
peso no	0.384000
desesperado hacer	0.344435
bajar peso	0.319055
...	...
etapa vida	0.000000
etapa super	0.000000
etapa depresión	0.000000
estupida vez	0.000000
zapatitos rojo	0.000000

En la tabla 19, se puede notar que solo ciertos bigramas tienen puntaje tf-idf, ya que son solo los que aparecen en este documento en particular. Todos los bigramas de este documento tienen una puntuación tf-idf y todo lo demás aparece como ceros, ya que cuanto más común sea el bigrama en todos los documentos, menor será su puntuación; por ejemplo, en este caso los bigramas “bajar peso” y “desesperado hacer” tienen una puntuación menor en comparación a los demás bigramas, esto quiere decir que estos bigramas están presentes en muchos más documentos. Además, cuanto más exclusivo sea un bigrama en el actual documento (por ejemplo, "hacer ejercicio" y "ejercicio deber"), mayor será la puntuación.

Se obtuvo las variables independientes y dependientes aplicando el sobremuestreo con SMOTE, para la variable independiente se usó el vector de características tf-idf obtenido y para la variable dependiente se usó la etiqueta sentimiento (0 para no depresivo y 1 para depresivo).

Finalmente, las variables dependientes e independiente obtenidas se almacenaron con el nombre de “x_tfidf_Bigrama” y “y_tfidf_Bigrama” respectivamente para uso posterior en el entrenamiento y prueba de los modelos.

6.2.2.3. TF-IDF para trigramas

- **Cálculo de la frecuencia de términos (TF)**

Mediante el módulo CountVectorizer se calculó el número de veces que aparece cada trigrama (3 palabras consecutivas) en cada uno de los documentos (tweets), y se guardó el vocabulario de todos los trigramas posibles con el nombre de “vocabulary_Trigrama.pkl” para uso posterior en posibles predicciones con nuevos datos, esto se realizó mediante el uso de la librería Pickle (ver sección 6.1.1.4.9). En la tabla 20 se muestra una previa de la tabla generada del recuento de todos los trigramas del conjunto de datos.

Tabla 20. Recuento de trigramas (TF) del conjunto de datos.

	abandonar serie duro	**abrace yo decir**	**abrazo depresion ansiedad**	**abri texto biblico**	**...**	**yo volver adicta**	**yo volviendo adicta**	**yo volviendo loco**	**you can do**
0	0	0	0	0	...	0	0	0	0
1	0	0	0	0	...	0	0	0	0
2	0	0	0	0	...	0	0	0	0
3	0	0	0	0	...	0	0	0	0
4	0	0	0	0	...	0	0	0	0
...	...	...	...	...	...	...	...	...	...
10709	0	0	0	0	...	0	0	0	0
10710	0	0	0	0	...	0	0	0	0
10711	0	0	0	0	...	0	0	0	0
10712	0	0	0	0	...	0	0	0	0
10713	0	0	0	0	...	0	0	0	0

El recuento de trigramas obtuvo como resultado 10714 filas, que es el total de documentos o tweets y 2321 columnas, es decir el total de trigramas únicos que están presentes en todo el dataset y que están presentes en al menos 2 documentos (min_df=2).

Como se puede ver en la tabla 20, al igual que con los unigramas y bigramas, la mayoría de los elementos en el vector de características son 0, por lo que se genera una matriz dispersa. Esto se debe a que el número de trigramas que aparecen en cada documento (tweet) es solo una pequeña parte de todo el conjunto de documentos; por lo tanto, hay muchos trigramas que no aparecen y se marcaran como 0.

- **Cálculo de los valores IDF**

Se calculó los valores IDF en base a los trigramas que se obtuvieron anteriormente. En la tabla 21 se muestra algunos valores IDF que se obtuvieron mediante un marco de datos de Python, estos valores se presentan ordenados de forma ascendente.

Tabla 21. Previa de los valores idf para trigramas del conjunto de datos

	valores_idf
yo dar ansiedad	5.768540
yo dar cuenta	6.408199
no yo gustar	6.782892
no poder dormir	6.813664
yo dar depresion	6.845413
...	...
huele bicho cabron	9.180788
hpta no poder	9.180788
hpta fumarme tabaco	9.180788
hoy yo llamar	9.180788
you can do	9.180788

Como se puede observar en la tabla 21, los trigramas “yo dar depresion” y “no poder dormir” tienen valores IDF más bajos en comparación con otros trigramas. Esto quiere decir que estos trigramas aparecen en más documentos de la colección, ya que cuanto menor sea el valor IDF del trigrama, menos única será para cualquier documento en particular.

- **Calculo de las puntuaciones TF-IDF**

Una vez que se calculó los valores IDF para trigramas, ya se puede calcular las puntuaciones tf-idf para cualquier documento (tweet) o conjunto de documentos.

Se realizó el cálculo de las puntuaciones tf-idf para todos los documentos del conjunto de datos. En la tabla 22 se presenta una previa de la representación de valores tf-idf para trigramas de un documento (tweet) que se colocó en un marco de datos para fines de visualización y se ordenaron sus puntajes en orden descendente.

Tabla 22. Valores tf-idf (trigramas) de un tweet del conjunto de datos

	tfidf
no hacer ejercicio	0.529491
hacer ejercicio deber	0.529491
peso no yo	0.529491
no yo dar	0.398647
persona sufrir ansiedad	0.000000
...	...
grave ver influencers	0.000000
grito desesperado no	0.000000
guayabo moral mañana	0.000000

gustar no yo	0.000000
you can do	0.000000

En la tabla 22 se puede notar que solo ciertos trigramas tienen puntaje tf-idf, ya que son solo los que aparecen en este documento en particular. Todos los trigramas de este documento tienen una puntuación tf-idf y todo lo demás aparece como ceros, ya que cuanto más común sea el trigrama en todos los documentos, menor será su puntuación; por ejemplo, en este caso el trigrama “no yo dar” tiene una puntuación menor en comparación a los demás trigramas, esto quiere decir que este trigrama está presente en más documentos. Además, cuanto más exclusivo sea un trigrama en el actual documento (por ejemplo, "no hacer ejercicio" y "hacer ejercicio deber"), mayor será la puntuación.

Se obtuvo las variables independientes y dependientes aplicando el sobremuestreo con SMOTE, para la variable independiente se usó el vector de características tf-idf obtenido y para la variable dependiente se usó la etiqueta sentimiento (0 para no depresivo y 1 para depresivo) del dataset.

Finalmente, las variables dependientes e independiente obtenidas se almacenaron con el nombre de “x_tfidf_Trigrama” y “y_tfidf_Trigrama” respectivamente para uso posterior en el entrenamiento y prueba de los modelos.

Todo el proceso de extracción de características para unigramas, bigramas y trigramas se lo realizó en Jupyter notebook, y se puede ver en detalle en el repositorio[10] .

6.2.3. Tarea: Detección de sentimiento

Fase 4. Minería de texto y construcción de hipótesis

Los vectores de características con N-gramas obtenidos en la tarea de extracción de características se usaron para realizar el entrenamiento con los 3 modelos de Machine learning, es decir, se usaron los vectores con unigramas, bigramas y trigramas para entrenar el modelo de Maquinas de Vectores de Soporte, Random Forest y Naive Bayes, y así determinar cuál de los 3 ofrece el mejor rendimiento en cada uno de los modelos.

Además, debido a la gran cantidad de tweets depresivos identificados en el preprocesamiento, se plantea la siguiente hipótesis: En tiempos de covid-19 existe un aumento en la cantidad de publicaciones depresivas en Twitter.

6.2.3.1. Modelo de Máquinas de Vectores de Soporte (SVM)

[10] https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/3.Extraccion_caracteristicas/ExtraccionCaracteristicas.ipynb

- **Máquinas de Vectores de Soporte con Unigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Unigrama”, y el archivo “y_tfidf_Unigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 15. Dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

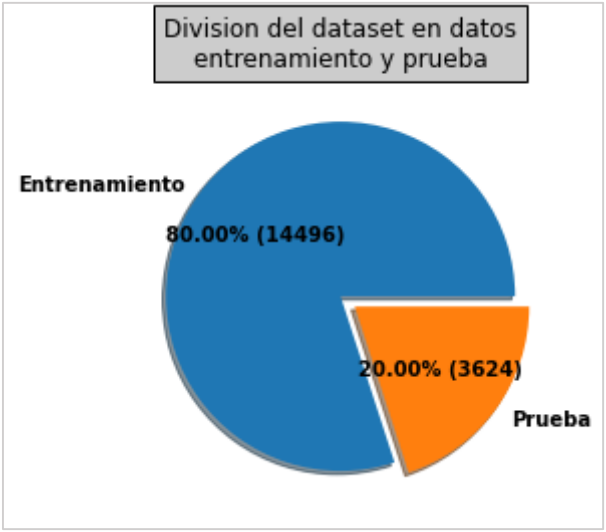


Figura 15. División de los datos en entrenamiento y prueba para el modelo SVM con Unigramas

A través de la librería Scikit-Learn se empleó el algoritmo Máquinas de Vectores de Soporte para la creación del modelo usando kernel = “linear” como hiperparámetro, ya que las características en clasificación de texto en general se organizan en categorías linealmente separables (ver sección 4.7.1).

Así mismo, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 23.

Tabla 23. Validación cruzada del modelo SVM con unigramas.

	Exactitud (Accuracy)
K=5	0.94896552
	0.94377372
	0.95584684
	0.94756813
	0.94963781
Exactitud media:	**0.949158**
Desviación estándar:	**0.003912**

El resultado de la validación cruzada da una precisión media de 95%, dicho resultado debería ser similar para el conjunto de prueba en caso de que los datos estén particionados adecuadamente, esto se muestra a continuación.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 24 y la figura 16 respectivamente.

Tabla 24. Reporte de clasificación para el modelo SVM con unigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.98	0.92	0.95	1779
1 (Contenido depresivo)	0.93	0.98	0.95	1845
Accuracy (exactitud)			0.95	3624
Macro avg	0.95	0.95	0.95	3624
Weighted avg	0.95	0.95	0.95	3624

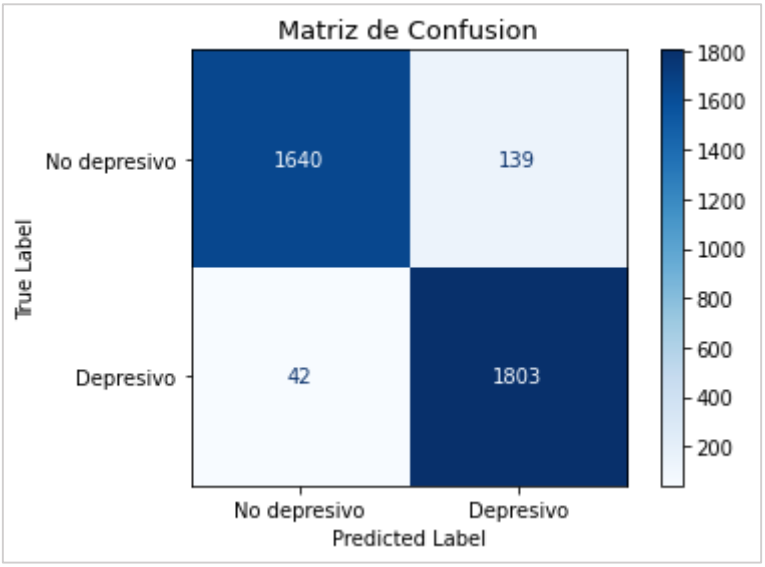


Figura 16. Matriz de confusión del modelo SVM con unigramas.

En la tabla 24 se puede ver que la exactitud en la predicción con el conjunto de datos de prueba es de 95%, por lo que se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es igualmente 95%; en consecuencia, se puede afirmar que el modelo es confiable para identificar contenido depresivo y no depresivo.

En cuanto a la matriz de confusión reflejada en la figura 16, se observa el número de publicaciones que fueron predichos de forma correcta (diagonal principal), y se puede notar

que la cantidad de valores respecto a los que no fueron clasificados de forma correcta por el modelo está acorde al porcentaje reflejado en el reporte de clasificación (tabla 24).

- **Máquinas de Vectores de Soporte con Bigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Bigrama”, y el archivo “y_tfidf_Bigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 17. Dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

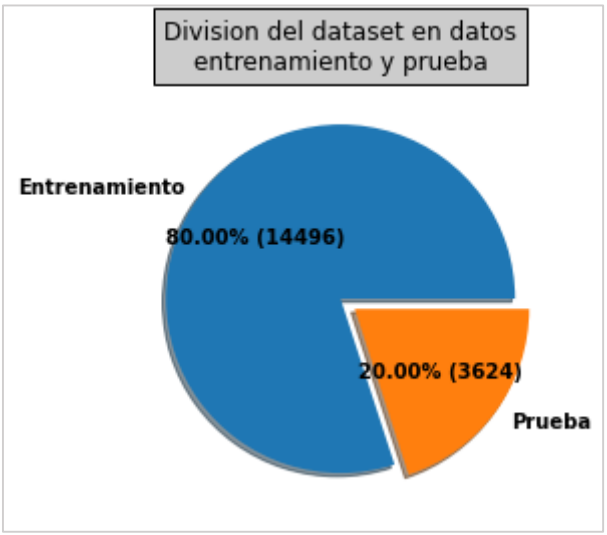


Figura 17. División de los datos en entrenamiento y prueba para el modelo SVM con Bigramas.

De la misma forma que se hizo para crear el modelo SVM con Unigramas, se usó la librería Scikit-Learn para emplear el algoritmo Máquinas de Vectores de Soporte a fin de crear del modelo usando kernel = “linear” como hiperparámetro.

Así mismo, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 25.

Tabla 25. Validación cruzada del modelo SVM con bigramas.

	Exactitud (Accuracy)
K=5	0.81965517
	0.82166264
	0.81993791
	0.82476716
	0.8361504
Exactitud media:	**0.824435**
Desviación estándar:	**0.006134**

El resultado de la validación cruzada da una precisión media de 82%, donde se aprecia que este modelo tuvo un rendimiento bastante menor respecto a los resultados del modelo con unigramas presentado anteriormente; sin embargo, se continúa ejecutando las siguientes pruebas para validar los resultados obtenidos.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 26 y la figura 18 respectivamente.

Tabla 26. Reporte de clasificación para el modelo SVM con bigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.81	0.86	0.84	1779
1 (Contenido depresivo)	0.86	0.81	0.83	1845
Accuracy (exactitud)			0.83	3624
Macro avg	0.84	0.84	0.83	3624
Weighted avg	0.84	0.83	0.83	3624

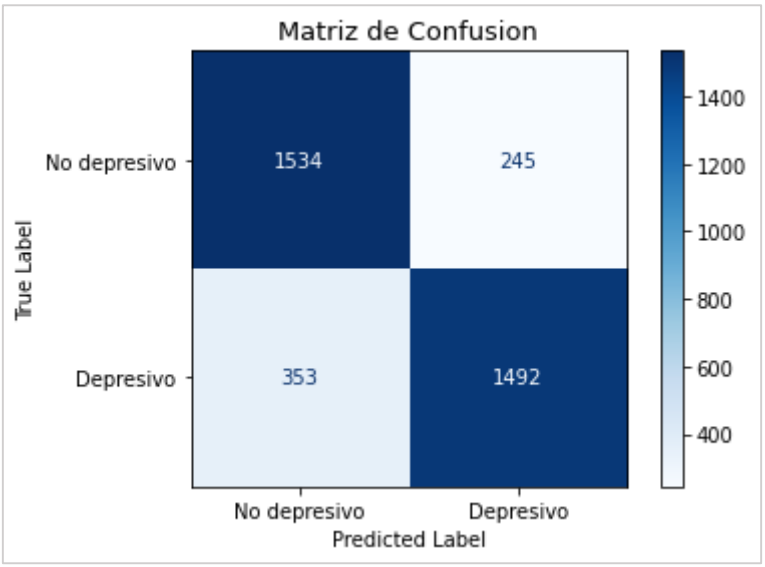


Figura 18. Matriz de confusión del modelo SVM con bigramas.

En la tabla 26 se ve que la exactitud en la predicción con el conjunto de datos de prueba es 83%, y se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es 82%, por lo que se puede afirmar que los resultados son confiables; sin embargo, los porcentajes obtenidos son bajos para ser aceptables en comparación con los resultados obtenidos con unigramas, esto se ve más claramente en la matriz de confusión reflejada en la figura 18, en donde se observa que la cantidad de valores predichos de forma correcta (diagonal principal) es más baja en comparación con las que no fueron clasificados correctamente (diagonal secundaria).

- **Máquinas de Vectores de Soporte con Trigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Trigrama”, y el archivo “y_tfidf_Trigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 19. Dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

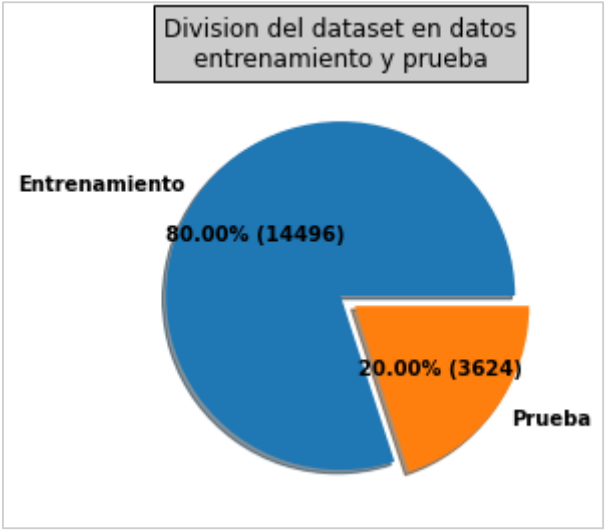


Figura 19. División de los datos en entrenamiento y prueba para el modelo SVM con Trigramas

De la misma forma que se hizo para crear el modelo SVM con Unigramas y Bigramas, se usó la librería Scikit-Learn para emplear el algoritmo Máquinas de Vectores de Soporte a fin de crear del modelo usando kernel = “linear” como hiperparámetro.

Así mismo, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 27.

Tabla 27. Validación cruzada del modelo SVM con trigramas.

	Exactitud (Accuracy)
K=5	0.63
	0.61952397
	0.61710935
	0.6115902
	0.6264229
Exactitud media:	**0.620929**
Desviación estándar:	**0.006576**

El resultado de la validación cruzada da una precisión media de 62%, donde se aprecia que este modelo tuvo un rendimiento bastante menor respecto a los resultados del modelo con unigramas y bigramas presentado anteriormente; sin embargo, se continúa ejecutando las siguientes pruebas para validar los resultados obtenidos.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 28 y la figura 20 respectivamente.

Tabla 28. Reporte de clasificación para el modelo SVM con trigramas.

	Precisión	Recall	F1-Score	Support
0 (Contenido no depresivo)	0.56	0.95	0.71	1779
1 (Contenido depresivo)	0.85	0.29	0.43	1845
Accuracy (exactitud)			0.61	3624
Macro avg	0.71	0.62	0.57	3624
Weighted avg	0.71	0.61	0.57	3624

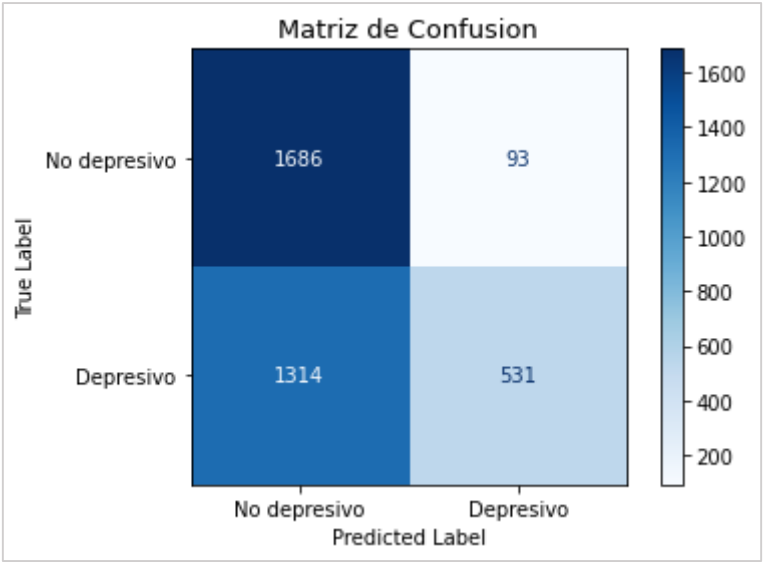


Figura 20. Matriz de confusión del modelo SVM con trigramas.

En la tabla 28 se ve que la exactitud en la predicción con el conjunto de datos de prueba es 61%, y se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es 62%, por lo que se puede afirmar que los resultados son confiables; sin embargo, los porcentajes obtenidos son bastante bajos para ser aceptables, incluso más que el resultado obtenido en la predicción con los bigramas mostrado anteriormente. Los resultados se ven más claramente en la matriz de confusión reflejada en la figura 20, en donde se observa que la cantidad de valores predichos de forma correcta (diagonal principal) es incluso menor en comparación con las que no fueron clasificados correctamente (diagonal secundaria), sobre todo en el caso del contenido clasificado como depresivo.

- **Rendimiento de los 3 modelos de Maquinas de Vectores de Soporte**

Para una mejor comprensión, se presenta la comparación en la figura 21 del rendimiento de los 3 modelos SVM con unigramas, bigramas y con trigramas a través del porcentaje obtenido en cada uno de ellos.

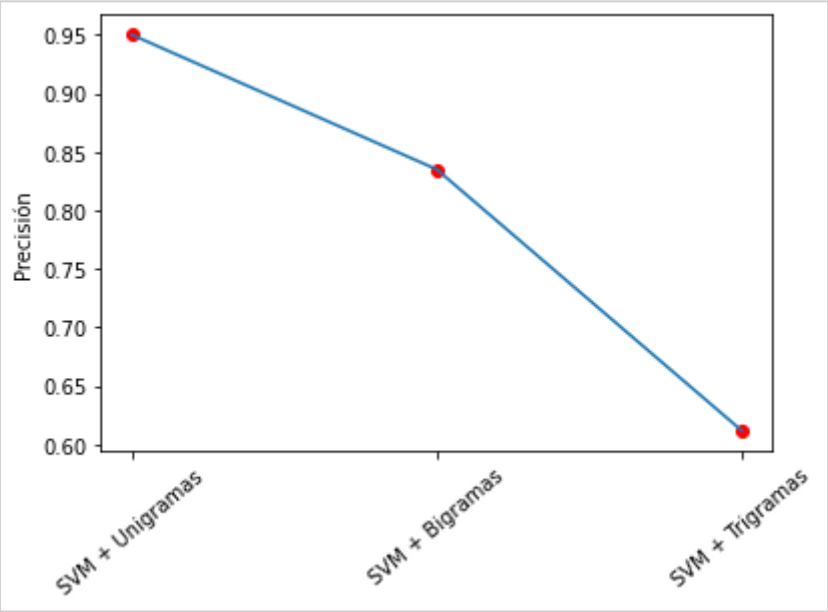


Figura 21. Comparación del rendimiento de los 3 modelos SVM.

Como se puede ver en la figura 21, el rendimiento de los 3 modelos de Maquinas de Vectores de Soporte (SVM) va decreciendo en su exactitud de forma casi lineal de acuerdo a su uso con unigramas, bigramas y trigramas respectivamente, siendo el modelo SVM con unigramas el que mejor porcentaje de rendimiento ofrece; por lo tanto, a este modelo con mejor rendimiento se lo exportó para uso posterior con el nombre “modelo_SVM_Unigram.pkl” mediante el uso de la librería Joblib (ver sección 6.1.1.4.8).

Todo el proceso de entrenamiento y prueba del modelo Maquinas de Vectores de Soporte con unigramas, bigramas y trigramas se lo puede ver en detalle en el repositorio[11].

6.2.3.2. Modelo de Random Forest

- **Random Forest con Unigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Unigrama”, y el archivo “y_tfidf_Unigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

[11] https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/Support_Vector_Machines.ipynb

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 22. Dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

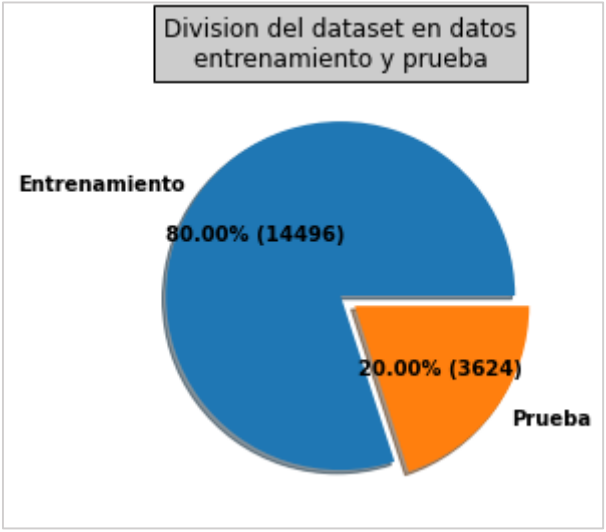


Figura 22. División de los datos en entrenamiento y prueba para el modelo RF con Unigramas.

Para entrenar el modelo Random Forest, fue necesario en primer lugar establecer la cantidad óptima de árboles de decisión para el modelo. De acuerdo a [74], una mayor cantidad de árboles solo aumenta su costo computacional y no tiene una ganancia significativa en el rendimiento y sugiere que un rango entre 64 y 128 árboles es suficiente para obtener resultados satisfactorios; no obstante, se ejecutó el modelo con 5 valores distintos para el número de árboles que están alrededor de la cantidad recomendada para comprobar el rendimiento con el conjunto de datos que se está usando, tal como se muestra en la figura 23.

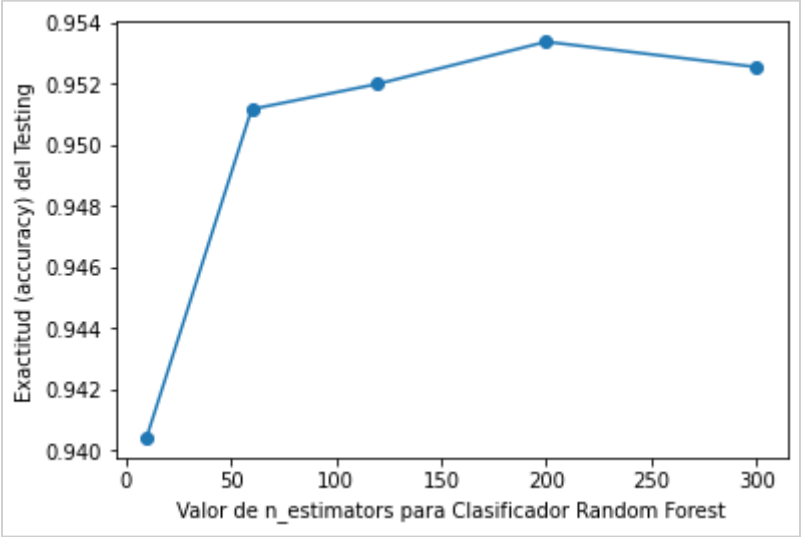


Figura 23. Rendimiento del modelo RF con unigramas en distintas cantidades de árboles.

Como se puede ver en la figura 23, el modelo Random Forest con unigramas logra un mejor rendimiento en el conjunto de prueba cuando el número de árboles esta alrededor de 200; por lo tanto, a través de la librería Scikit-Learn se empleó el algoritmo Random Forest para la creación del modelo usando el hiperparámetro “n_estimators=200”.

Además, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 29.

Tabla 29. Validación cruzada del modelo RF con unigramas.

	Exactitud (Accuracy)
K=5	0.95068966
	0.94687823
	0.95343222
	0.95239738
	0.954812
Exactitud media:	**0.951642**
Desviación estándar:	**0.002736**

El resultado de la validación cruzada da una precisión media de 95%, dicho resultado debería ser similar para el conjunto de prueba en caso de que los datos estén particionados adecuadamente, esto se muestra a continuación.
Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 30 y la figura 24 respectivamente.

Tabla 30. Reporte de clasificación para el modelo RF con unigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.98	0.93	0.95	1779
1 (Contenido depresivo)	0.93	0.98	0.96	1845
Accuracy (exactitud)			0.95	3624
Macro avg	0.96	0.95	0.95	3624
Weighted avg	0.96	0.95	0.95	3624

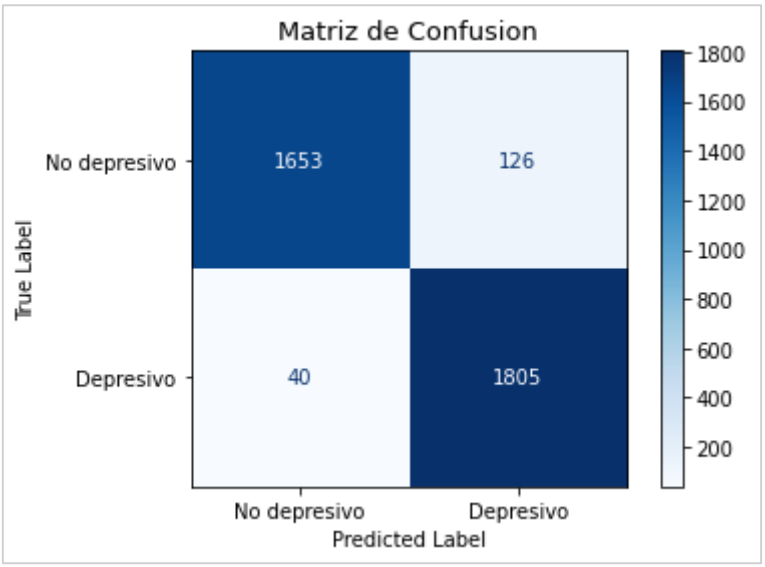


Figura 24. Matriz de confusión del modelo RF con unigramas.

En la tabla 30 se puede ver que la exactitud en la predicción con el conjunto de datos de prueba es de 95%, por lo que se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es igualmente 95%, en consecuencia, se puede afirmar que el modelo es confiable para identificar contenido depresivo y no depresivo. En cuanto a la matriz de confusión reflejada en la figura 24, se observa el número de publicaciones que fueron predichos de forma correcta (diagonal principal), y se puede notar que la cantidad de valores respecto a los que no fueron clasificados de forma correcta por el modelo está acorde al porcentaje reflejado en el reporte de clasificación (tabla 30).

- **Random Forest con Bigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Bigrama”, y el archivo “y_tfidf_Bigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 25. Dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

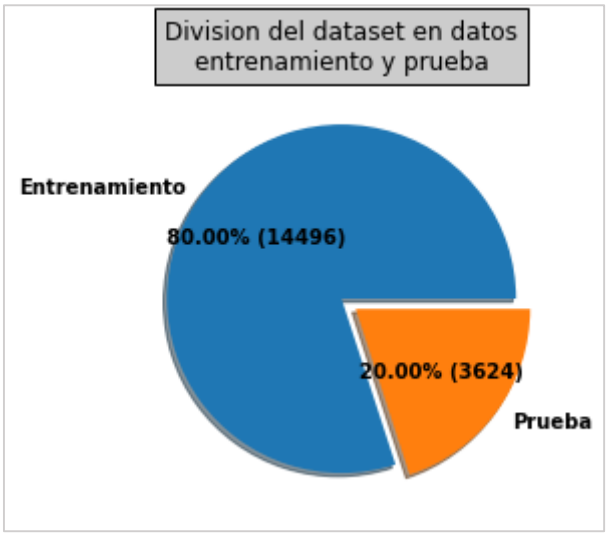


Figura 25. División de los datos en entrenamiento y prueba para el modelo RF con Bigramas.

Para entrenar el modelo Random Forest, al igual que en el caso del modelo con unigramas, primero se buscó determinar la cantidad óptima de árboles de decisión para el modelo, por ende, se ejecutó el modelo con 4 valores distintos para el número de árboles para comprobar el rendimiento con el conjunto de datos que se está usando, tal como se muestra en la figura 26.

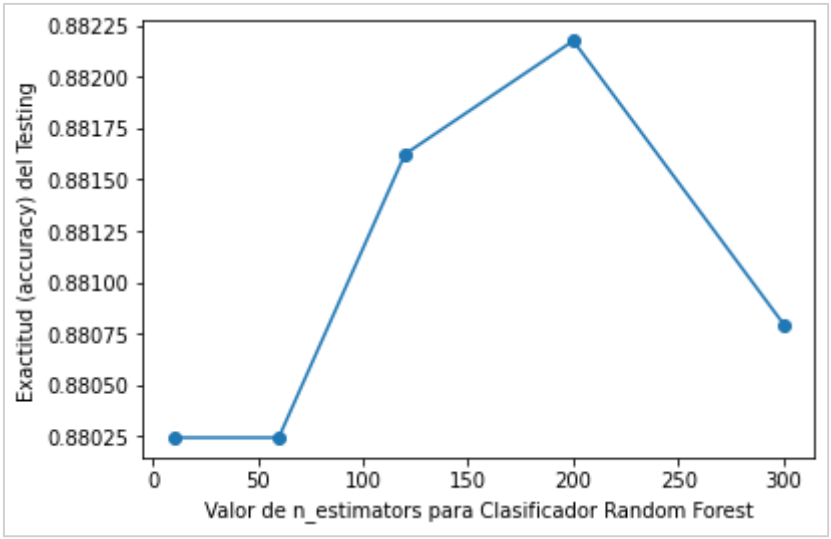


Figura 26. Rendimiento del modelo RF con bigramas en distintas cantidades de árboles.

Como se puede ver en la figura 26, el modelo Random Forest con bigramas logra un mejor rendimiento en el conjunto de prueba cuando el número de árboles esta alrededor de 200; por lo tanto, a través de la librería Scikit-Learn se empleó el algoritmo Random Forest para la creación del modelo usando el hiperparámetro “n_estimators=200”.

Además, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 31.

Tabla 31. Validación cruzada del modelo RF con bigramas.

	Exactitud (Accuracy)
K=5	0.88517241
	0.87685409
	0.88168334
	0.88202829
	0.88237323
Exactitud media:	**0.881622**
Desviación estándar:	**0.002686**

El resultado de la validación cruzada da una precisión media de 88%, donde se aprecia que este modelo tuvo un rendimiento bastante menor respecto a los resultados del modelo con unigramas presentado anteriormente; sin embargo, se continúa ejecutando las siguientes pruebas para validar los resultados obtenidos.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 32 y la figura 27 respectivamente.

Tabla 32. Reporte de clasificación para el modelo RF con bigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.88	0.89	0.88	1779
1 (Contenido depresivo)	0.89	0.88	0.89	1845
Accuracy (exactitud)			0.88	3624
Macro avg	0.88	0.88	0.88	3624
Weighted avg	0.88	0.88	0.88	3624

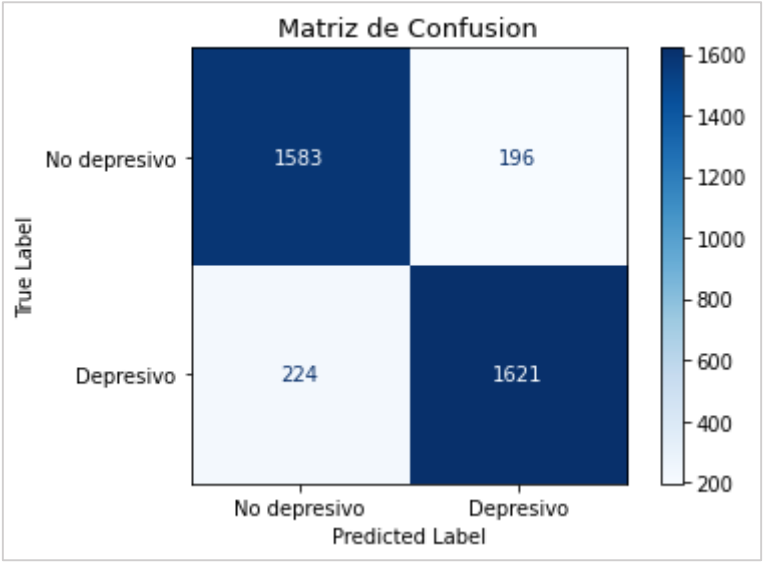


Figura 27. Matriz de confusión del modelo RF con bigramas.

En la tabla 32 se ve que la exactitud en la predicción con el conjunto de datos de prueba es 88%, y se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es igualmente 88%, por lo que se puede afirmar que los resultados son confiables; sin embargo, los porcentajes obtenidos son bajos para ser aceptables en comparación con los resultados obtenidos con unigramas, esto se ve más claramente en la matriz de confusión reflejada en la figura 27, en donde se observa que la cantidad de valores predichos de forma correcta (diagonal principal) es más baja en comparación con las que no fueron clasificados correctamente (diagonal secundaria).

- **Random Forest con Trigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Trigrama”, y el archivo “y_tfidf_Trigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 28, dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

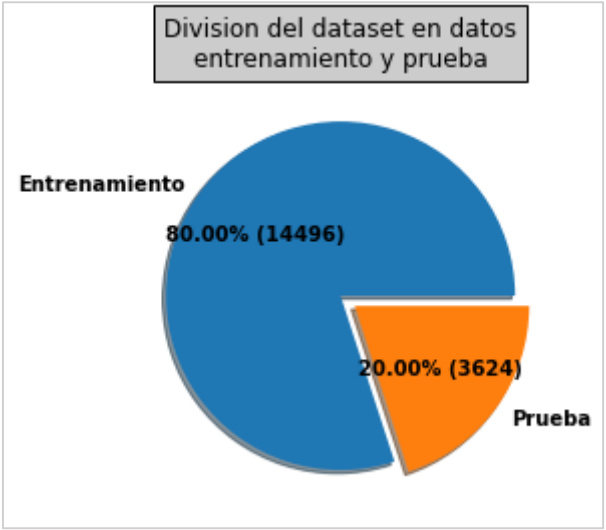


Figura 28. División de los datos en entrenamiento y prueba para el modelo RF con Trigramas.

De la misma forma que se hizo para crear el modelo RF con Unigramas y Bigramas, primero se buscó determinar la cantidad óptima de árboles de decisión para el modelo, por ende, se ejecutó el modelo con 5 valores distintos para el número de árboles para comprobar el rendimiento con el conjunto de datos que se está usando, tal como se muestra en la figura 29.

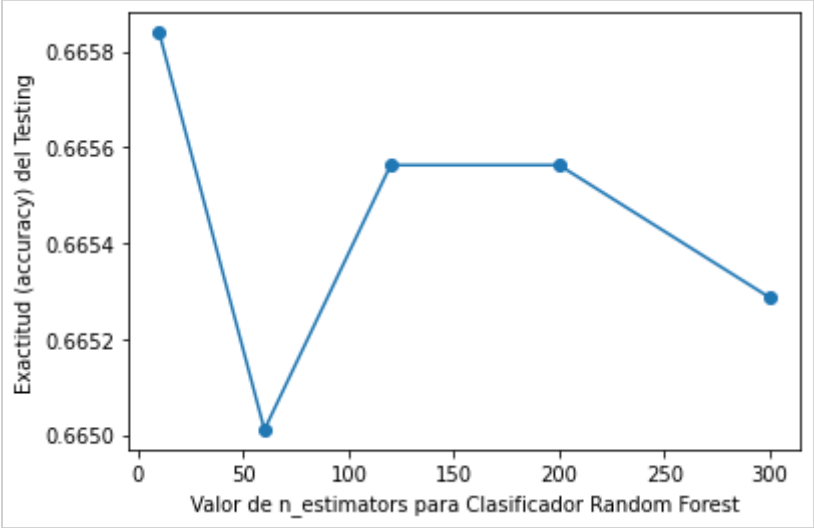


Figura 29. Rendimiento del modelo RF con trigramas en distintas cantidades de árboles.

Como se puede ver en la figura 29, el modelo Random Forest con trigramas tiene una variación casi insignificante con las distintas cantidades de árboles en las que se ejecutó, sin embargo, logra un rendimiento un poco más alto cuando el número de árboles esta alrededor de 10; por lo tanto, a través de la librería Scikit-Learn, se empleó el algoritmo Random Forest para la creación del modelo usando el hiperparámetro “n_estimators=10”.

Así mismo, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección

4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 33.

Tabla 33. Validación cruzada del modelo RF con trigramas.

	Exactitud (Accuracy)
K=5	0.66965517
	0.66367713
	0.66747154
	0.65643325
	0.67747499
Exactitud media:	**0.666942**
Desviación estándar:	**0.006924**

El resultado de la validación cruzada da una precisión media de 66%, donde se aprecia que este modelo tuvo un rendimiento bastante menor respecto a los resultados del modelo con unigramas y bigramas presentado anteriormente; sin embargo, se continúa ejecutando las siguientes pruebas para validar los resultados obtenidos.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 34 y la figura 30 respectivamente.

Tabla 34. Reporte de clasificación para el modelo RF con trigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.60	0.95	0.74	1779
1 (Contenido depresivo)	0.89	0.39	0.54	1845
Accuracy (exactitud)			0.67	3624
Macro avg	0.74	0.67	0.64	3624
Weighted avg	0.75	0.67	0.64	3624

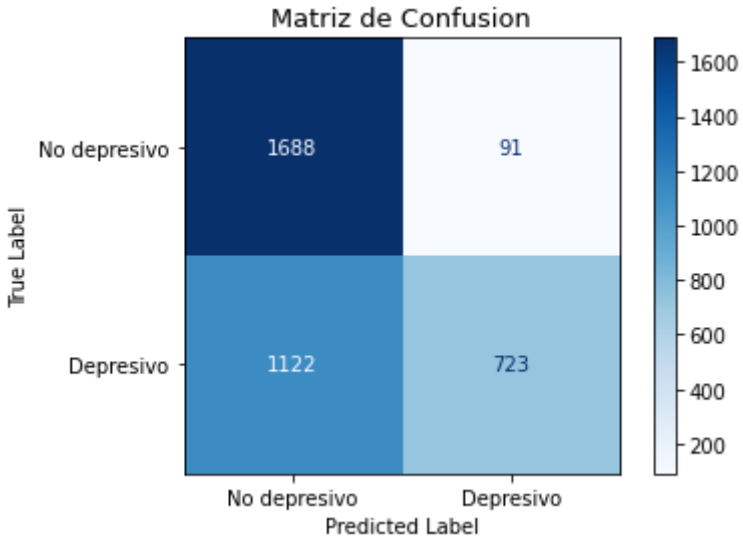


Figura 30. Matriz de confusión del modelo RF con trigramas.

En la tabla 34 se ve que la exactitud en la predicción con el conjunto de datos de prueba es 67%, y se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es 66%, por lo que se puede afirmar que los resultados son confiables; sin embargo, los porcentajes obtenidos son bastante bajos para ser aceptables, incluso siendo más bajos que el resultado obtenido en la predicción en el modelo con bigramas mostrado anteriormente. Los resultados se ven más claramente en la matriz de confusión reflejada en la figura 30, en donde se observa que la cantidad de valores predichos de forma correcta (diagonal principal) es menor en comparación con las que no fueron clasificados correctamente (diagonal secundaria), sobre todo en el caso del contenido clasificado como depresivo.

- **Rendimiento de los 3 modelos de Random Forest.**

Para una mejor comprensión, se presenta la comparación en la figura 31 del rendimiento de los 3 modelos Random Forest con unigramas, bigramas y trigramas a través del porcentaje obtenido en cada uno de ellos.

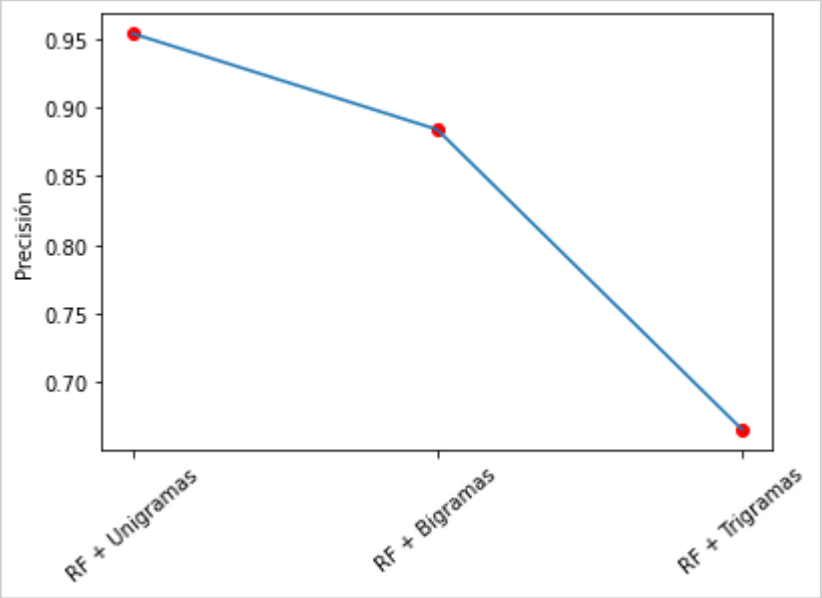


Figura 31. Comparación del rendimiento de los 3 modelos Random Forest.

Como se puede ver en la figura 31, el rendimiento de los 3 modelos de Random Forest (RF) va decreciendo en su exactitud de acuerdo a su uso con unigramas, bigramas y trigramas respectivamente, siendo el modelo RF con unigramas el que mejor porcentaje de rendimiento ofrece; por lo tanto, a este modelo con mejor rendimiento se lo exportó para uso posterior con el nombre "modelo_RF_Unigram.pkl" mediante el uso de la librería Joblib. (ver sección 6.1.1.4.8).

Todo el proceso de entrenamiento y prueba del modelo Random Forest con unigramas, bigramas y trigramas se lo puede ver en detalle en el repositorio[12].

6.2.3.3. Modelo de Naive Bayes

- **Naive Bayes con Unigramas**

Para la ejecución del presente algoritmo, se importó el vector de características "x_tfidf_Unigrama", y el archivo "y_tfidf_Unigrama" para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 32, dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

[12] https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/Random_Forest.ipynb

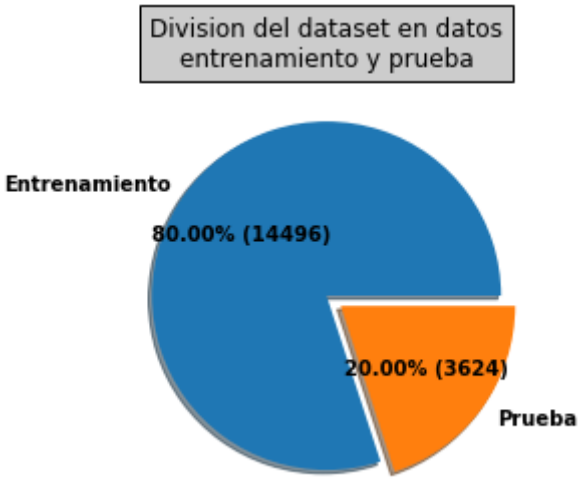


Figura 32. División de los datos en entrenamiento y prueba para el modelo NB con Unigramas.

A través de la librería Scikit-Learn, se empleó el algoritmo Naive Bayes para la creación del modelo. Además, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 35.

Tabla 35. Validación cruzada del modelo NB con unigramas.

	Exactitud (Accuracy)
K=5	0.89310345
	0.87478441
	0.89582615
	0.88030355
	0.8851328
Exactitud media:	**0.885830**
Desviación estándar:	**0.007821**

El resultado de la validación cruzada da una precisión media de 88%, dicho resultado debería ser similar para el conjunto de prueba en caso de que los datos estén particionados adecuadamente, esto se muestra a continuación.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 36 y la figura 33 respectivamente.

Tabla 36. Reporte de clasificación para el modelo NB con unigramas.

	Precisión	Recall	F1-Score	Support
0 (Contenido no depresivo)	0.96	0.83	0.89	1779
1 (Contenido depresivo)	0.85	0.96	0.90	1845
Accuracy (exactitud)			0.90	3624
Macro avg	0.90	0.89	0.90	3624
Weighted avg	0.90	0.90	0.90	3624

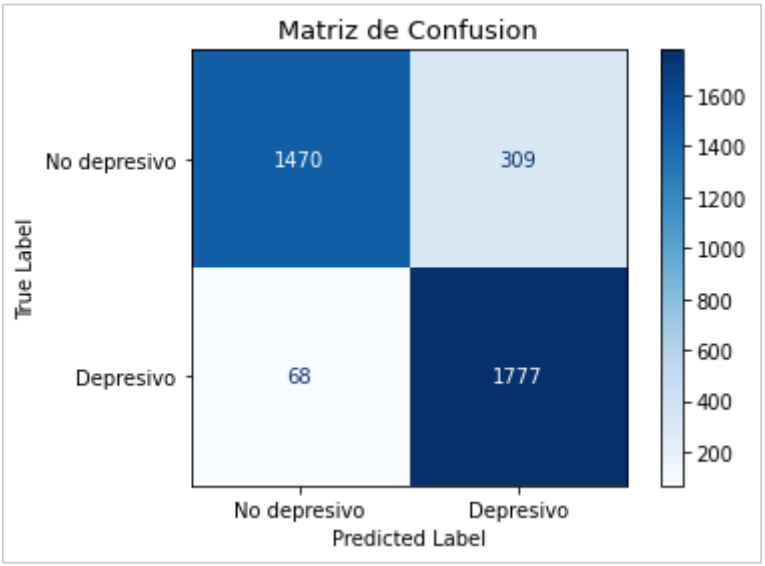


Figura 33. Matriz de confusión del modelo NB con unigramas.

En la tabla 36 se puede ver que la exactitud en la predicción con el conjunto de datos de prueba es de 90%, por lo que se puede comprobar que los resultados están muy apegados a la exactitud media de la validación cruzada que es 88%; en consecuencia, se puede afirmar que el modelo es confiable para identificar contenido depresivo y no depresivo. En cuanto a la matriz de confusión reflejada en la figura 33, se observa el número de publicaciones que fueron predichos de forma correcta (diagonal principal), y se puede distinguir que la cantidad de valores respecto a los que no fueron clasificados de forma correcta por el modelo está acorde al porcentaje reflejado en el reporte de clasificación (tabla 36).

- **Naive Bayes con Bigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Bigrama”, y el archivo “y_tfidf_Bigrama” para obtener las etiquetas (sentimiento) del

conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 34, dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

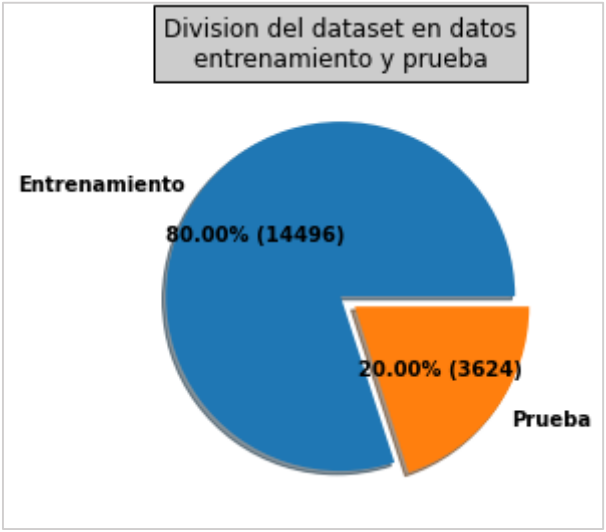


Figura 34. División de los datos en entrenamiento y prueba para el modelo NB con Bigramas.

De igual manera que se hizo para crear el modelo NB con Unigramas, se usó la librería Scikit-Learn para emplear el algoritmo Naive Bayes a fin de crear del modelo. Así mismo, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 37.

Tabla 37. Validación cruzada del modelo NB con bigramas.

	Exactitud (Accuracy)
K=5	0.82
	0.81407382
	0.82373232
	0.82511211
	0.82718179
Exactitud media:	**0.822020**
Desviación estándar:	**0.004612**

El resultado de la validación cruzada da una precisión media de 82%, donde se aprecia que este modelo tuvo un rendimiento menor respecto a los resultados del modelo con unigramas presentado anteriormente, sin embargo, se continúa ejecutando las siguientes pruebas para validar los resultados obtenidos.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 38 y la figura 35 respectivamente.

Tabla 38. Reporte de clasificación para el modelo NB con bigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.88	0.79	0.84	1779
1 (Contenido depresivo)	0.82	0.90	0.86	1845
Accuracy (exactitud)			0.85	3624
Macro avg	0.85	0.85	0.85	3624
Weighted avg	0.85	0.85	0.85	3624

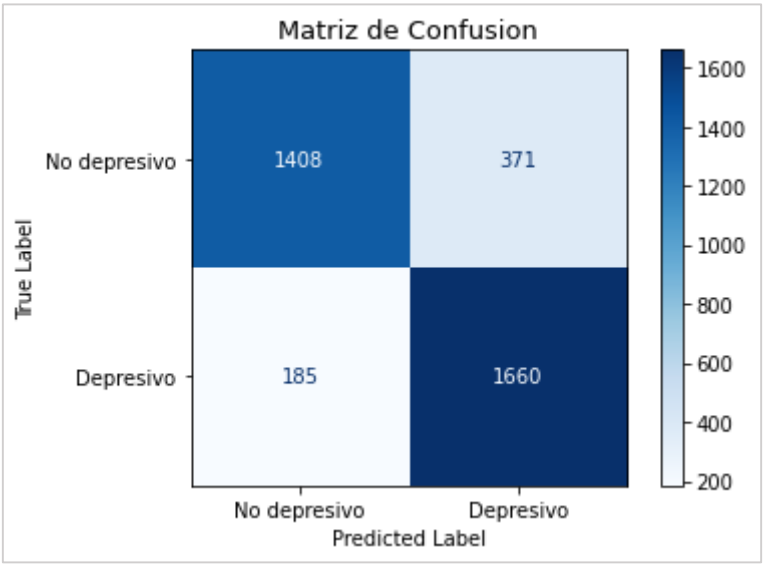


Figura 35. Matriz de confusión del modelo NB con bigramas.

En la tabla 38 se ve que la exactitud en la predicción con el conjunto de datos de prueba es 85%, y se puede comprobar que los resultados están cercanos a la exactitud media de la validación cruzada que es 82%, por lo que se puede afirmar que los resultados son confiables; sin embargo, los porcentajes obtenidos son bajos para ser aceptables en comparación con los resultados obtenidos con unigramas, esto se ve más claramente en la matriz de confusión reflejada en la figura 35, en donde se observa que la cantidad de valores predichos de forma

correcta (diagonal principal) es más baja en comparación con los que no fueron clasificados correctamente (diagonal secundaria).

- **Naive Bayes con Trigramas**

Para la ejecución del presente algoritmo, se importó el vector de características “x_tfidf_Trigrama”, y el archivo “y_tfidf_Trigrama” para obtener las etiquetas (sentimiento) del conjunto de datos. Estos archivos representan a las variables independientes y dependientes respectivamente.

Los datos se dividieron en entrenamiento y prueba en proporción de 80% para entrenamiento y 20% para prueba como se muestra en la figura 36. Dichas muestras se tomaron de forma aleatoria mediante la librería Scikit-Learn (ver sección 6.1.1.4.7).

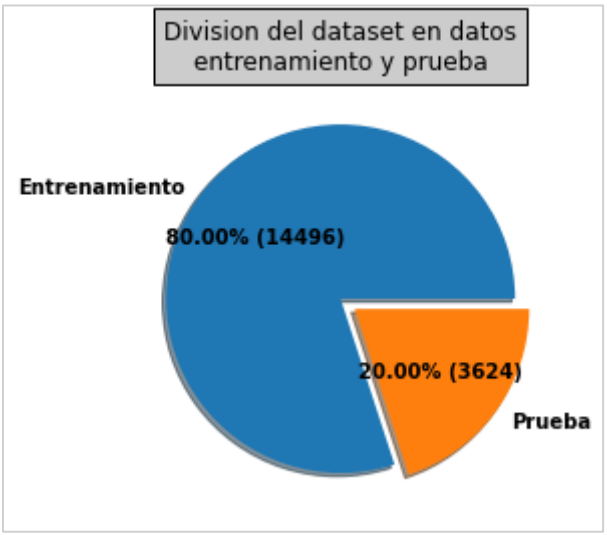


Figura 36. División de los datos en entrenamiento y prueba para el modelo NB con Trigramas.

De la misma forma que se hizo para crear el modelo NB con Unigramas y Bigramas, se usó la librería Scikit-Learn para emplear el algoritmo Naive Bayes a fin de crear del modelo. Así mismo, para garantizar que el rendimiento del modelo sea independiente de la partición entre datos de entrenamiento y prueba, se utilizó la técnica de validación cruzada (ver sección 4.12) a los datos de entrenamiento con 5 pliegues (K=5), cuyos resultados se pueden ver en la tabla 39.

Tabla 39. Validación cruzada del modelo NB con trigramas.

	Exactitud (Accuracy)
K=5	0.67137931
	0.6664367
	0.66367713
	0.65746809
	0.67506037
Exactitud media:	**0.666804**
Desviación estándar:	**0.006102**

El resultado de la validación cruzada da una precisión media de 66%, donde se aprecia que este modelo tuvo un rendimiento bastante menor respecto a los resultados del modelo con unigramas y bigramas presentado anteriormente, sin embargo, se continúa ejecutando las siguientes pruebas para validar los resultados obtenidos.

Se generó el reporte de clasificación para medir la calidad de las predicciones con el conjunto de datos de prueba, además de la respectiva matriz de confusión, éstos se pueden ver en la tabla 40 y la figura 37 respectivamente.

Tabla 40. Reporte de clasificación para el modelo NB con trigramas.

	Precisión	**Recall**	**F1-Score**	**Support**
0 (Contenido no depresivo)	0.60	0.94	0.73	1779
1 (Contenido depresivo)	0.87	0.40	0.54	1845
Accuracy (exactitud)			0.66	3624
Macro avg	0.73	0.67	0.64	3624
Weighted avg	0.74	0.66	0.64	3624

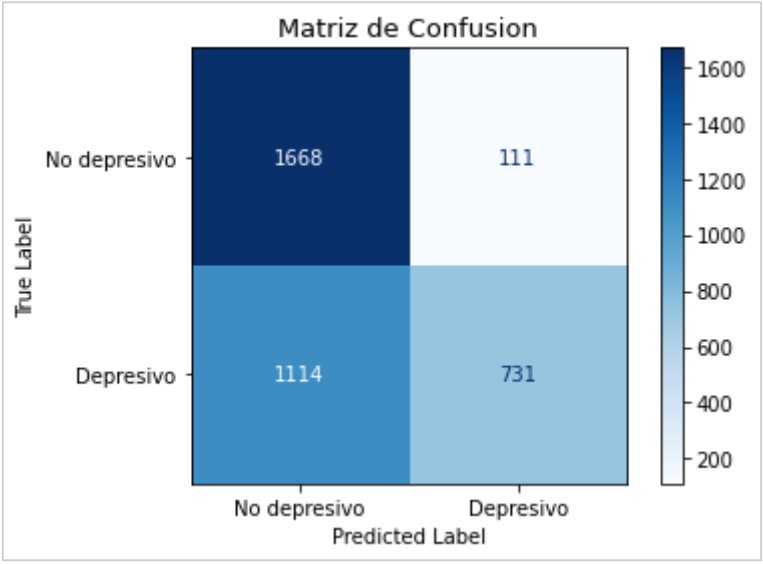


Figura 37. Matriz de confusión del modelo NB con trigramas.

En la tabla 40 se ve que la exactitud en la predicción con el conjunto de datos de prueba es 66%, y se puede comprobar que los resultados están cercanos a la exactitud media de la validación cruzada que es igualmente 66%, por lo que se puede afirmar que los resultados son confiables; sin embargo, los porcentajes obtenidos son bastante bajos para ser aceptables, incluso más que el resultado obtenido en la predicción con los bigramas mostrado anteriormente. Los resultados se ven más claramente en la matriz de confusión reflejada en la figura 37, en donde se observa que la cantidad de valores predichos de forma correcta (diagonal principal) es incluso menor en comparación con las que no fueron clasificados correctamente (diagonal secundaria), sobre todo en el caso del contenido clasificado como depresivo.

- **Rendimiento de los 3 modelos de Naive Bayes.**

Para una mejor comprensión, se presenta la comparación en la figura 38 del rendimiento de los 3 modelos Naive Bayes con unigramas, bigramas y trigramas a través del porcentaje obtenido en cada uno de ellos.

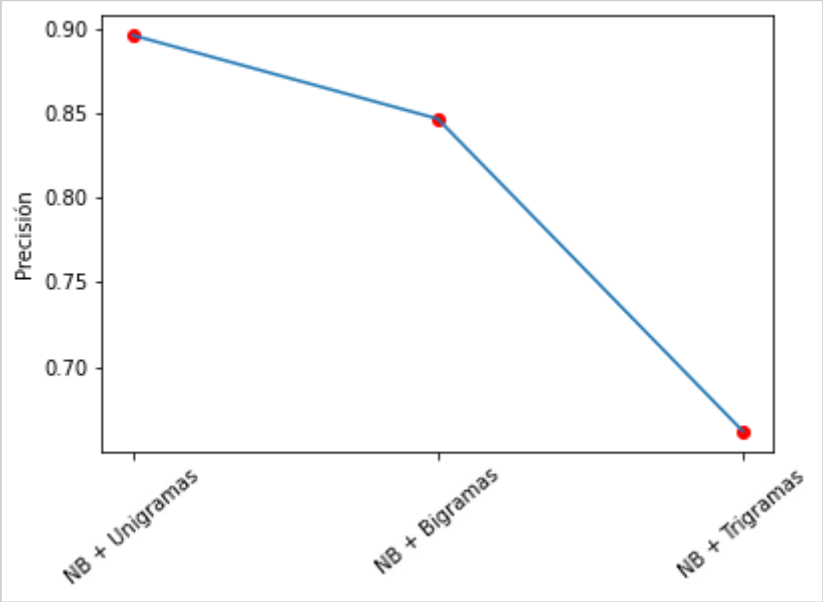


Figura 38. Comparación del rendimiento de los 3 modelos Naive Bayes.

Como se puede ver en la figura 38, el rendimiento de los 3 modelos Naive Bayes (NB) va decreciendo en su exactitud de forma significativa de acuerdo a su uso con unigramas, bigramas y trigramas respectivamente, siendo el modelo Naive Bayes con unigramas el que mejor porcentaje de rendimiento ofrece; por lo tanto, a este modelo con mejor rendimiento se lo exportó para uso posterior con el nombre “modelo_NB_Unigram.pkl” mediante el uso de la librería Joblib. (ver sección 6.1.1.4.8).

Todo el proceso de entrenamiento y prueba del modelo Naive Bayes con unigramas, bigramas y trigramas se lo puede ver en detalle en el repositorio[13].

[13] https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/Naive_Bayes.ipynb

6.3. Objetivo 3: Interpretar los resultados obtenidos en el análisis de sentimientos

Fase 5. Interpretación/evaluación

En esta sección se realizó el análisis en base a los mejores modelos que fueron obtenidos mediante el entrenamiento en el objetivo 2, mediante las métricas de rendimiento y en base a la cantidad de predicciones realizadas por cada uno de los modelos, además se realizó una predicción a los tweets del 2019 mediante el mejor modelo obtenido para finalmente realizar un análisis univariado y bivariado a los conjuntos de datos obtenidos.

6.3.1. Tarea: Evaluar el desempeño de los algoritmos mediante métricas de precisión, accuracy, recall y F1 Score.

Se utilizó el modelo con mejor rendimiento de acuerdo a cada tipo de algoritmo, es decir el mejor modelo de Maquinas de vectores de soporte, Random Forest y Naive Bayes para comparar el rendimiento de estos 3 modelos. Cabe recalcar que los modelos con mejor rendimiento para identificar contenido depresivo fueron los que se entrenaron con unigramas y que se obtuvieron en el objetivo 2.

A través de la aplicación de estos 3 modelos de algoritmos de clasificación, se realizó la predicción con el conjunto de datos de prueba, en la figura 39 se muestran la cantidad de tweets del conjunto de datos de prueba por cada clase (depresivo y no depresivo). Cabe recalcar que es el mismo conjunto de datos de prueba para los 3 modelos por lo que se puede observar que tienen la misma cantidad de valores. Mientras que en la figura 40 se muestra la cantidad de tweets que fueron predichos por el modelo de Maquinas de vectores de soporte, Random Forest y Naive Bayes respectivamente.

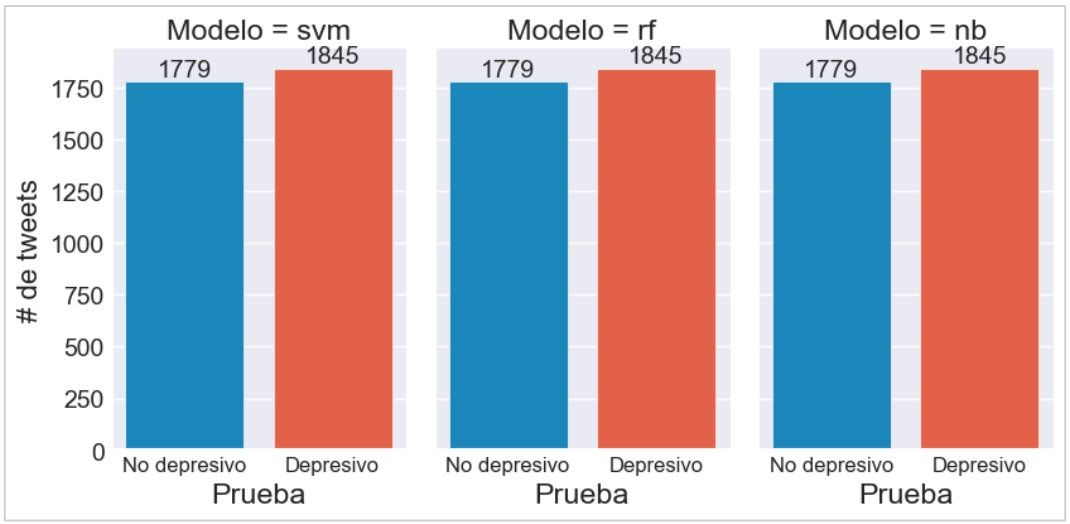


Figura 39. Cantidad de tweets del conjunto de datos de prueba por cada clase.

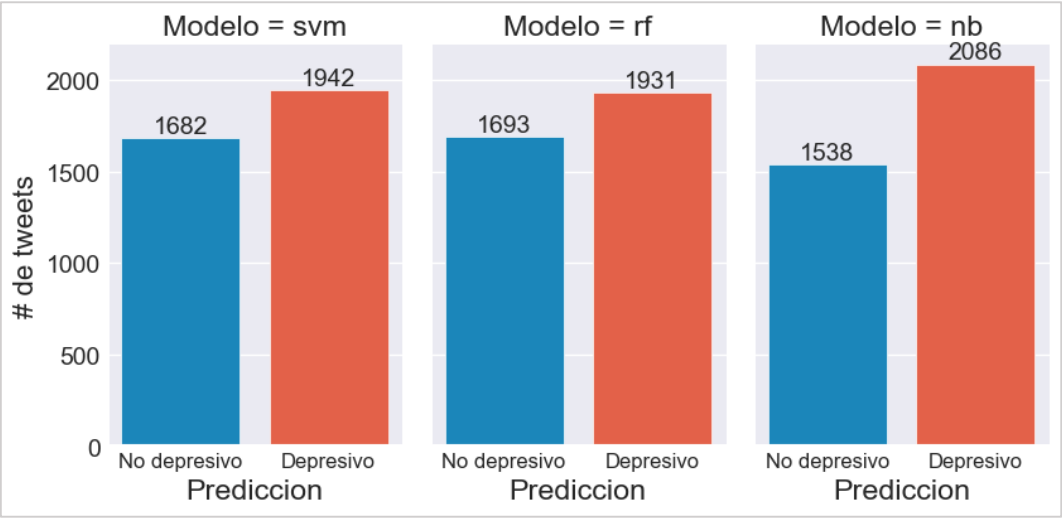
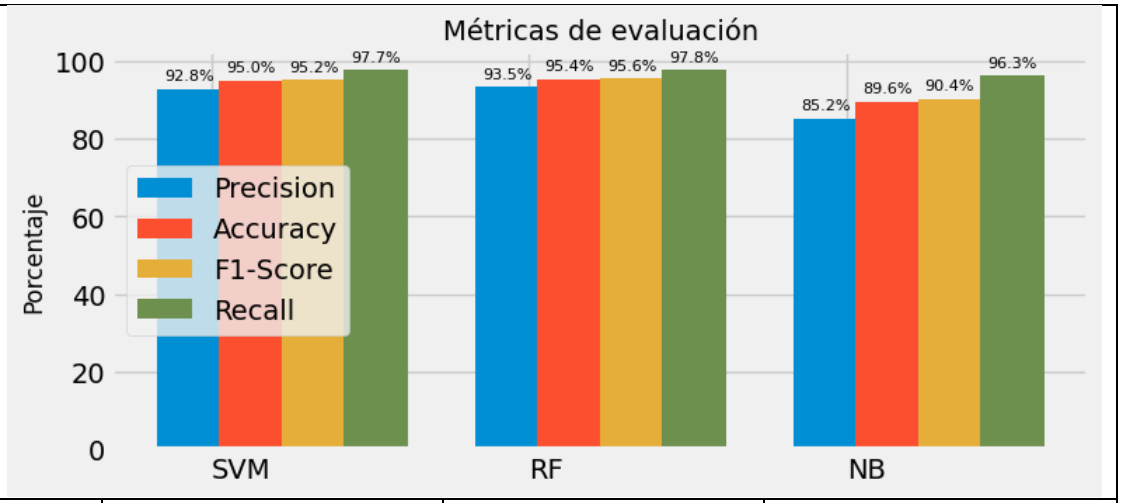


Figura 40. Cantidad de tweets predichos por cada modelo.

En la figura 40 se puede ver la cantidad de valores predichos por los modelos de Maquinas de vectores de soporte y Random Forest respecto al conjunto de prueba (figura 39) es bastante cercano, mientras que en el modelo Naive Bayes se ve una variabilidad más notable en la cantidad de datos predichos.

Para tener una mejor perspectiva del rendimiento de los 3 modelos, en la tabla 41 se presenta una comparación del rendimiento de cada uno de ellos utilizando las métricas de Precisión, Exactitud(Accuracy), Puntuación F1 y Recall.

Tabla 41. Comparación de los modelos finales mediante métricas de evaluación.



	Support Vector Machines	Random Forest	Naive Bayes
Precision	0.9284	0.9347	0.8518
Accuracy	0.9500	0.9541	0.8959
F1	0.9522	0.9560	0.9040
Recall	0.9772	0.9783	0.9631

Como se puede ver en la tabla 41, el modelo de Random Forest supera a los otros 2 modelos en todas las métricas mostradas, con valores:

- Precisión = 93.4%
- Exactitud (accuracy) = 95.4%
- Puntuación F1 = 95.6%
- Recall = 97.8%

El modelo de Maquinas de Vectores de Soporte (Support Vector Machines) está detrás con una diferencia casi insignificante, estos valores son:

- Precisión = 92.8%
- Exactitud (accuracy) = 95%
- Puntuación F1 = 95.2%
- Recall = 97.7%

El modelo Naive Bayes tiene el rendimiento más bajo de los 3 modelos, pero aun con valores relativamente altos, estos son:

- Precisión = 85.2%
- Exactitud (accuracy) = 89.5%
- Puntuación F1 = 90.4%
- Recall = 96.3%

En base al análisis de las métricas de evaluación mostradas, el modelo Random Forest fue seleccionado como el mejor para la clasificación de contenido depresivo, siguiéndole muy de cerca el modelo de Maquinas de Vectores de Soporte, y finalmente el modelo Naive Bayes como el que menor rendimiento tiene entre los 3 modelos.

La comparación de los modelos entrenados se la realizó en Jupyter y se puede ver en mayor detalle en el repositorio [14].

[14] https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/4.Comparacion_y_Analisis_Modelos/1.Comparacion_Analisis_Mejores_Modelos.ipynb

6.3.2. Tarea: Predecir contenido depresivo con tweets prepandemia

A través del modelo que tuvo mejor rendimiento para identificar contenido depresivo en twitter y que fue identificado en la tarea anterior, se realizó la predicción para tweets que se han publicado previo al inicio del covid-19, específicamente del año 2019. Para esto, se extrajeron tweets de todo el año 2019 utilizando las mismas características que se emplearon para extraer los tweets depresivos usados para el entrenamiento (ver sección 6.1.3), obteniendo un total de 12125 tweets.

Los tweets extraídos se preprocesaron siguiendo el mismo procedimiento que se realizó para el preprocesamiento de los datos que fueron usados en el entrenamiento de los modelos (ver sección 6.2.1) con excepción de la limpieza manual, esto con el fin de que se pueda usar con distintos datos nuevos sin necesidad de realizar procedimientos manuales. Con los tweets preprocesados se creó un nuevo vector tf-idf usando el vocabulario guardado de los unigramas (ver sección 6.2.2.1) para que el nuevo vector creado tenga la misma longitud de características que tienen los datos entrenados. Cabe mencionar que se usó el vocabulario de los unigramas porque es el que se utilizó para el modelo que tuvo mejor rendimiento.

Por último, se realizó la predicción de tweets depresivos utilizando el modelo con mejor rendimiento guardado con el nombre de “modelo_RF_Unigram”. Algunos tweets que fueron clasificados como depresivos por este modelo se presentan en la tabla 42.

Tabla 42. Tweets del 2019 clasificados como depresivos por el modelo RF.

Tweet Original	Tweet Preprocesado	Predicción
Ya les conté que la ansiedad me está matando ? Pues buej, aquí me tienen 😁	['conte', 'ansiedad', 'yo', 'matar', 'pues', 'buej', 'aqui', 'yo', 'cara_radiante_con_ojos_sonrientes']	Depresivo
No me aguanto este estrés y ansiedad que cargo.	['no', 'yo', 'aguantar', 'estre', 'ansiedad', 'cargo']	Depresivo
Siento como la hora de la depresión llega poco a poco	['sentir', 'hora', 'depresion', 'llegar']	Depresivo
A veces la depresión y la ansiedad no me deja ver lo que en realidad las personas hacen por mí.	['vez', 'depresion', 'ansiedad', 'no', 'yo', 'dejar', 'ver', 'realidad', 'persona', 'hacer']	Depresivo
Ni el psicólogo me ayuda con mi situación, de seguro y sale loco o sumergido en la depresión conmigo y me acompaña a llorar.	['ni', 'psicologo', 'yo', 'ayuda', 'situacion', 'seguro', 'salir', 'loco', 'sumergido', 'depresion', 'yo', 'yo', 'acompañar', 'llorar']	Depresivo

Los tweets clasificados como depresivos por el modelo se guardaron en un conjunto de datos con el nombre de “Tweets_Depresivos_2019_Predichos”. Todo el proceso llevado a cabo para extraer y preprocesar los tweets del año 2019, además del proceso de clasificación de los tweets usando el modelo Random Forest se puede ver en el repositorio [15].

6.3.3. Tarea: Realizar análisis univariado y bivariado de los datos para representar y comparar la cantidad de tuits depresivos.

Para el presente análisis se utilizó el dataset que contiene tweets depresivos y que fue limpiado en la fase de preprocesamiento (ver sección 6.2.1), además del dataset que se obtuvo mediante la predicción con el modelo Random Forest (ver sección 6.3.2), para realizar la respectiva comparación. Ambos dataset tienen algunos atributos (columnas) en común que se crean al momento de la extracción de los tweets y que serán usados para realizar el análisis de éstos. En la tabla 43 se puede ver un resumen de los campos que contienen ambos dataset.

Tabla 43. Atributos del conjunto de datos con tweets depresivos.

Nombre de atributo	Descripción	Ejemplo
Id	Contiene un número identificador distinto para cada tweet	1339912665586610000
Date	Contiene la fecha de publicación del tweet	2020-12-18
Time	Contiene la hora de la publicación del tweet	07:37:28
Tweet	Contiene la redacción del tweet	a veces es válido sentirse agobiada desesperada y con una incertidumbre tenaz
geo	Contiene las coordenadas geográficas de donde se publicó el tweet (latitud, longitud y radio en km)	-1.3695361279507605, -78.08139447223985, 27.98571148655604km

Zonas geográficas donde hay más tweets depresivos

En primer lugar, mediante el dataset que contiene tweets depresivos que se limpió en la fase de preprocesamiento (tweets en tiempos de covid-19), se utilizó el atributo llamado “geo” que

15 https://github.com/byronmb/Identificacion_Depresion_Ecuador/tree/main/4.Comparacion_y_Analisis_Modelos

contiene la latitud, longitud y radio en kilómetros de la ubicación geográfica que fue usada para recolectar determinados tweets.

El atributo “geo” se separó en campos individuales que contienen Latitud, Longitud y Radio respectivamente para graficar las zonas en donde existen una mayor distribución de tweets depresivos en el Ecuador, tal como se muestra en la figura 41.

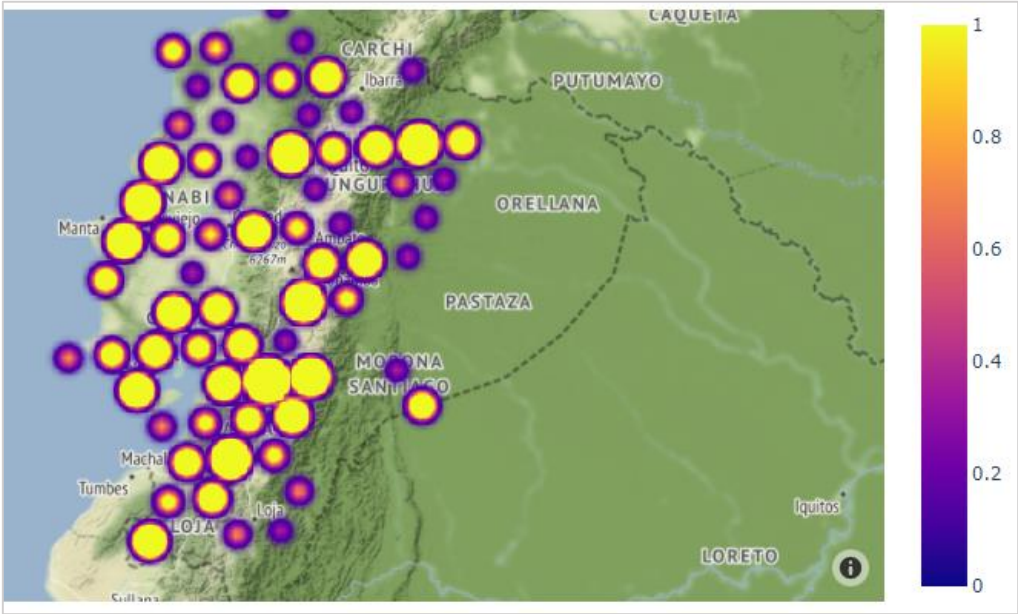


Figura 41. Mapa de calor de la distribución de tweets depresivos en Ecuador.

En la figura 41 se puede ver que hay una mayor cantidad de publicaciones depresivas en Twitter en la región Costa y Sierra, con presencia notoria en ciudades como Cuenca, Riobamba, Quito, Manabí y Loja, mientras que en la región Amazónica existe una cantidad relativamente baja en cuanto a publicaciones, con cierta presencia evidente en la provincia de Morona Santiago. Cabe recalcar que las coordenadas geográficas están relacionadas a las que se usaron para extraer los tweets (ver anexo 3), por lo que existen varios tweets que tienen la misma ubicación geográfica.

Distribución de publicaciones depresivas en base a las horas del día

En base al atributo “time” del dataset (ver tabla 43), se obtuvo la hora para conocer en qué horas del día existen una mayor o menor cantidad de tweets depresivos en el año 2020 y 2021, tal como se muestra en la figura 42.

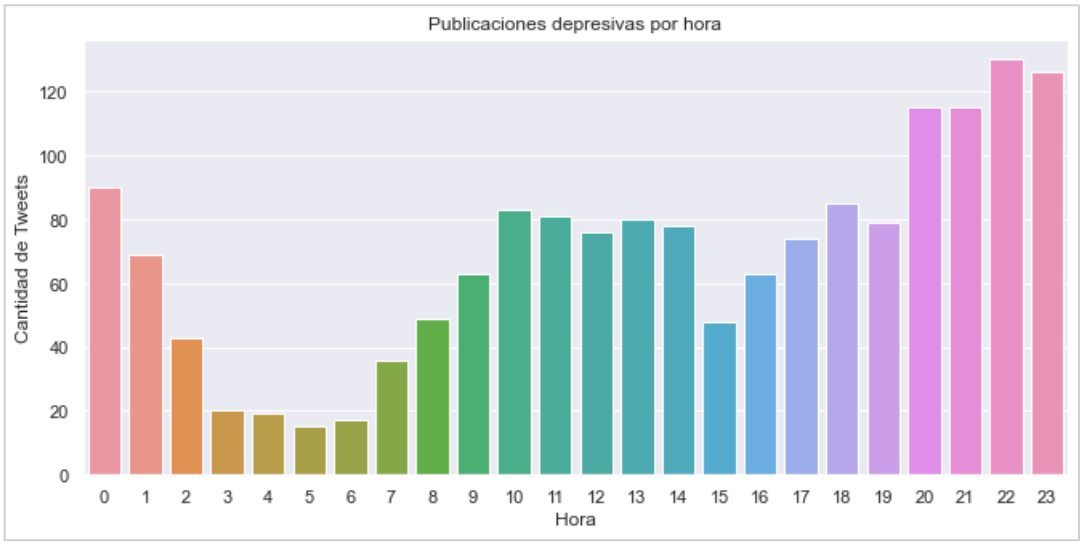


Figura 42. Distribución de publicaciones depresivas por hora.

Como se puede ver en la figura 42, las horas donde existen más publicaciones relacionadas a depresión son en la noche, teniendo una mayor cantidad entre las 20h00 y 00h00, mientras que en horas de la madrugada aproximadamente entre las 3h00 y 6h00 existe una menor cantidad de publicaciones depresivas.

Distribución de publicaciones depresivas en base a los meses de los años 2020 y 2021.

En base al atributo “date” del dataset (ver tabla 43), se separó el campo para obtener los meses y años en campos individuales para graficar la distribución de publicaciones por cada mes de los años 2020 y 2021, como se muestra en la figura 43.

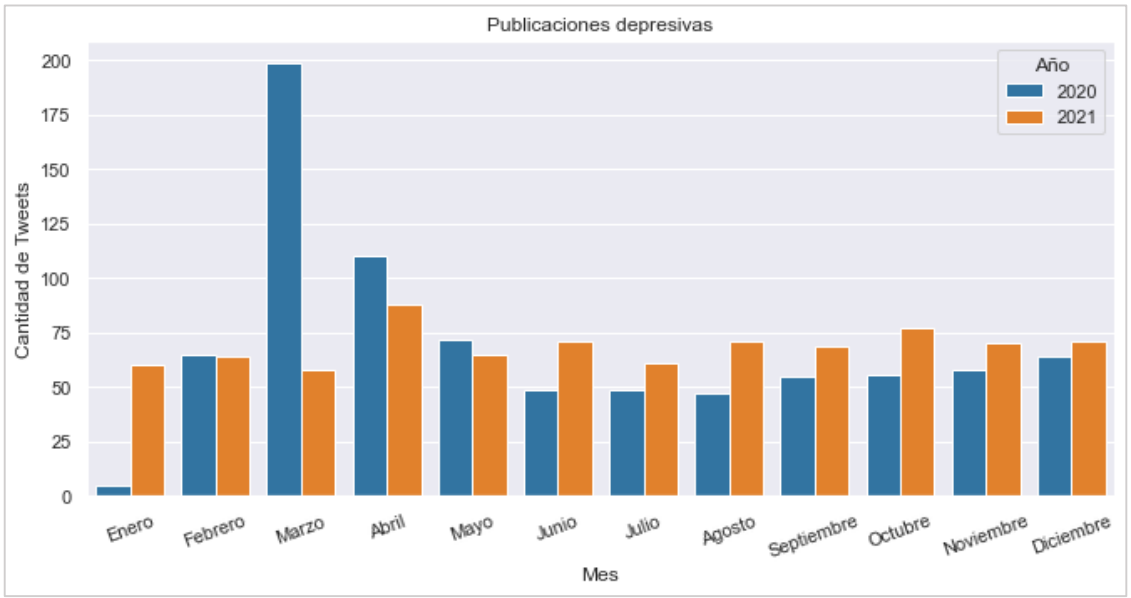


Figura 43. Publicaciones depresivas por meses de los años 2020 y 2021.

Como se puede ver en la figura 43, en el mes de marzo de 2020 es donde existe una mayor cantidad de publicaciones depresivas, esto concuerda con el inicio del confinamiento decretado en Ecuador por el coronavirus el 11 de marzo del 2020 [75], como segundo mes con mayor cantidad de publicaciones está el mes de abril del mismo año. Además, en los meses posteriores se puede ver una cantidad más baja pero relativamente constante en la cantidad de publicaciones sin presentar mayor variación. Sin embargo, se puede notar que en estos meses la cantidad de publicaciones del año 2021 son mayores que la cantidad de publicaciones del año 2020.

Cabe recalcar que el mes de enero del 2020 no se muestra la representación de publicaciones ya que se recolectaron a partir del 30 de enero del 2020 (ver sección 6.1.2).

Variación porcentual en las publicaciones depresivas en prepandemia y durante la pandemia

Se utilizó el dataset que se obtuvo mediante la predicción con el modelo Random Forest (ver sección 6.3.2) junto al dataset con tweets depresivos obtenido en la fase de preprocesamiento (ver sección 6.2.1) para comparar la cantidad de publicaciones en base a cada año. En la figura 44 se puede ver la cantidad de publicaciones depresivas del año 2019, 2020 y 2021 respectivamente con su valor y representación porcentual.

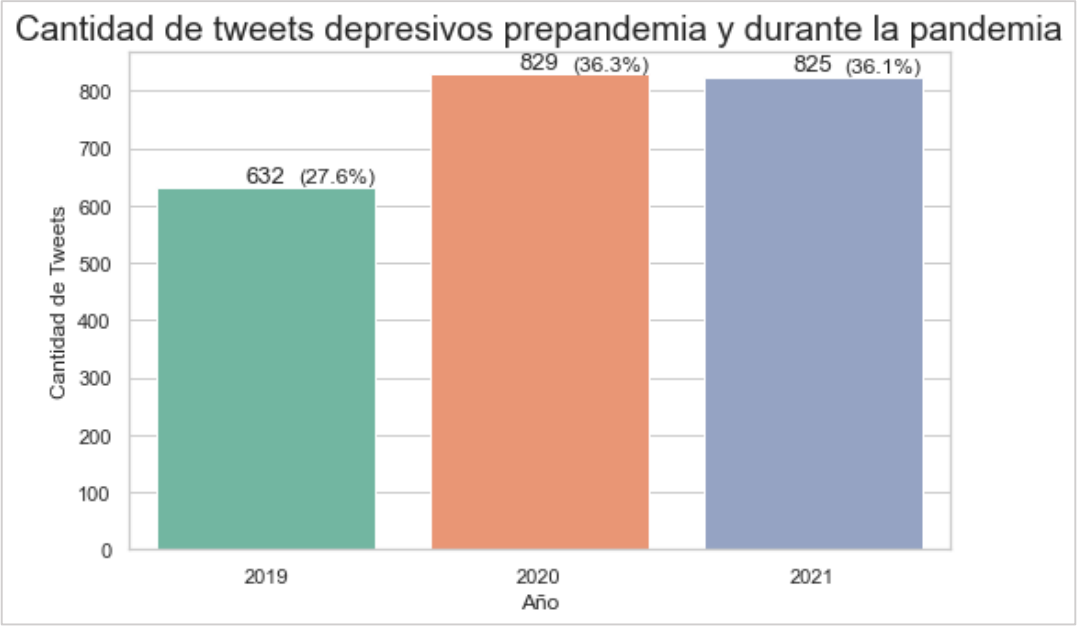


Figura 44. Cantidad de tweets depresivos de los años 2019, 2020 y 2021

Como se puede ver en la figura 44, si se compara a los 3 años, el 2019 (publicaciones prepandemia), 2020 y 2021 (publicaciones durante la pandemia), se puede notar que existe un incremento considerable en la cantidad de publicaciones depresivas realizadas en twitter en los años 2020 y 2021.

En el 2019 existe un total de 632 tweets representando un 27.6% del total de los 3 años, en el año 2021 existen 829 publicaciones que representa un 36.3%, y en el año 2021 hay un total de 825 publicaciones, que representan un 36,1 % del total de los 3 años. Entre los años 2020 y 2021 podemos ver una diferencia relativamente baja en la cantidad de publicaciones depresivas, siendo el año 2020 el que tiene una cantidad un poco más alta. Sin embargo, si se compara el crecimiento desde el año 2019 hasta el 2020, existe una mayor diferencia en cuanto a la cantidad de publicaciones depresivas.

Para conocer el porcentaje de diferencia entre las publicaciones del año 2019 y 2020 se usó la siguiente formula:

$$\frac{|\text{valor nuevo} - \text{valor antiguo}|}{|\text{valor antiguo}|} \text{x}100\%$$

$$\frac{829 - 632}{632} x100\% = 31.17\%$$

En base a la ecuación anterior, se puede decir que las publicaciones depresivas en twitter tuvieron un porcentaje de crecimiento de 31.17% en el año 2020 respecto a la cantidad de publicaciones del mismo tipo en 2019. Además, con esto se da contestación a la hipótesis planteada en la fase de Minería de texto y construcción de hipótesis (ver sección 6.2.3), con lo cual, si existe un incremento de publicaciones depresivas en tiempos de covid-19.

Todas las gráficas del análisis de pueden ver en detalle en el repositorio [16].

16 https://github.com/byronmb/Identificacion_Depresion_Ecuador/tree/main/4.Comparacion_y_Analisis_Modelos

7. Discusión

El desarrollo del presente Trabajo de Titulación titulado “Análisis de Sentimientos en Twitter para la Identificación de Depresión en Tiempos de COVID-19 en Ecuador ”, se lo realizo en 5 fases de la metodología KDT a través de los 3 objetivos planteados, a continuación, se describe cada uno de los objetivos para comprobar el cumplimiento de los mismos.

7.1. Objetivo 1: Construir un conjunto de datos a partir de las publicaciones de Twitter.

En esta fase se realizó una revisión de literatura mediante la cual se pudo detectar herramientas útiles para realizar un adecuado análisis de sentimientos, sobre todo para texto en español, ya que algunos conjuntos de herramientas existentes solo son compatibles con pocos idiomas, además de estar poco optimizadas en precisión (ver sección 6.1.1.4.4), asimismo de acuerdo a varios autores se pudo determinar cuáles son las mejores fases para realizar el análisis de sentimientos (ver sección 6.1.1.2) y de esta manera se logró comprobar que la metodología KDT se adapta mejor a las fases para realizar un adecuado análisis de sentimientos.

Posteriormente se extrajeron publicaciones de twitter en un rango desde el 30 de enero del 2020 hasta el 31 de diciembre del 2021 (tiempos de covid-19), mediante lo cual se pudo corroborar que se pueden obtener una gran cantidad de publicaciones que fueron útiles para el objeto de estudio, sin embargo también se comprobó que debido al uso de mapa de mosaicos H3 para obtener toda la zona geográfica de ecuador (ver sección 6.1.3), el radio de búsqueda se ajusta a cada celda, por lo que existen áreas en donde muchas publicaciones son recopiladas más de una vez, por lo tanto, hay que hacer una correcta eliminación de publicaciones duplicadas en el preprocesamiento.

7.2. Objetivo 2: Aplicar el análisis de sentimientos mediante una técnica basada en Machine Learning.

Este objetivo se llevó a cabo mediante el preprocesamiento de datos, extracción de características para culminar con el entrenamiento de datos de acuerdo a las fases de la metodología KDT.

Para realizar un adecuado preprocesamiento de los tweets recolectados se siguió una serie de fases con el objetivo de obtener un conjunto de datos optimo y que sea adecuado para ser fuente de entrenamiento en fases posteriores. Mediante esto se pudo identificar que twitter contiene gran cantidad de publicaciones con texto informal que además incluyen muchos usos idiosincrásicos, por lo que todos estos factores se tomaron en cuenta al momento de

desarrollar cada una de las fases del preprocesamiento con el objeto de que se pueda obtener datos lo más limpios posibles que sirvan para una clasificación óptima del sentimiento (ver sección 6.2.1).

Posteriormente, se graficó las 2 clases de tweets recolectados (tweets depresivos y tweets aleatorios) para comparar la cantidad de tweets de cada una de las clases, en donde el conjunto de datos con tweets depresivos se redujo considerablemente ya que fue filtrado con la ayuda de la Dra. Sandra Otero, Magister en Psicología Clínica Infantojuvenil[17] para filtrar los datos, manteniendo solo los tweets que indican depresión emocional, con lo cual se percibió un gran desequilibrio en comparación con la clase de tweets aleatorios; por lo tanto, se aplicó la técnica de sobremuestreo para equilibrar los datos

Luego del preprocesamiento de los datos se realizó la extracción de características de los tweets utilizando la técnica tf-idf, esto se cumplió de 3 formas para cada algoritmo, específicamente con unigramas, bigramas y trigramas, todo esto con el objetivo de comparar los resultados en el entrenamiento y comprobar el que mejor rendimiento ofrece.

Para culminar este objetivo, se realizó el entrenamiento de los datos, donde se pudo comprobar que el rendimiento de los modelos con los 3 distintos algoritmos (Maquinas de vectores de soporte, Random Forest y Naive Bayes), coinciden en brindar un mejor rendimiento cuando están entrenados con unigramas, disminuyendo este rendimiento de forma constante si se usan con bigramas o trigramas respectivamente, por lo tanto esto permite comprobar que los tweets no siguen un patrón similar en cuanto a la redacción en distintas publicaciones y así reafirmando la característica informal de estos. Cabe recalcar que para garantizar que el rendimiento de los modelos sea independiente de la partición de los datos de entrenamiento y prueba se utilizó la técnica de la validación cruzada obteniendo resultados que no discrepan demasiado en su precisión satisfactorios, tal como se constata en la sección 6.2.3.

[17] https://drive.google.com/drive/folders/1WKKCqOgM_PI1aYCbVBTIHoH6-4Zu8vhW?usp=sharing

7.3. Objetivo 3: Interpretar los resultados obtenidos en el análisis de sentimientos.

Al realizar este objetivo se pudo comprobar de forma gráfica los resultados de rendimiento de los mejores modelos entrenados en el objetivo 2, obteniendo como mejor modelo al Random Forest con una puntuación F1 del 95.6%, este resultado se puede contrastar con los trabajos relacionados [39], en donde se identificó a BERT como el mejor modelo de predicción para el análisis de sentimientos de las publicaciones, y en [40] obtuvieron un mejor rendimiento usando el modelo CNN-LSTM, en la figura 45 se presentan los porcentajes de rendimiento basado en la puntuación F1 por parte de los trabajos relacionados y del presente TT.

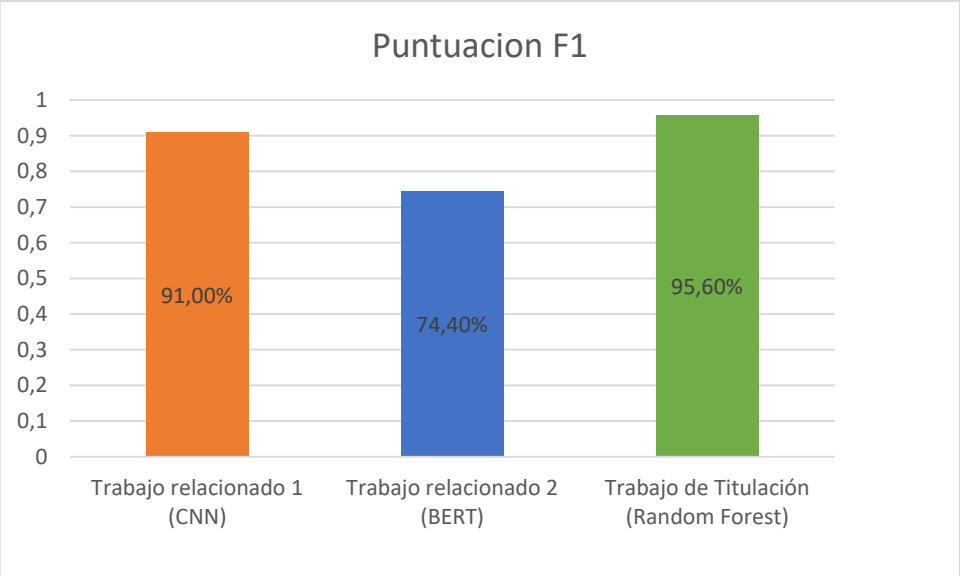


Figura 45 Comparación del rendimiento de modelos con trabajos relacionados..

En la figura se puede ver que los resultados del presente proyecto superan en cuanto al rendimiento con los trabajos relacionados, cabe mencionar que los trabajos relacionados usan otros modelos como del CNN (Convolutional neural networks) y BERT, y que son publicaciones en inglés, a diferencia del usado en el presente proyecto que es el modelo Random Forest y con publicaciones en español.

Con el mejor modelo para clasificar contenido depresivo se predijeron tweets de tiempos prepandemia, específicamente del año 2019, en donde se pudo dar contestación a la pregunta de investigación: ¿En qué medida existen tuits con contenido depresivo en Twitter en tiempos de covid-19 en Ecuador?. Se comparó los datos del 2019 con los extraídos del año 2020 y 2021 (tiempos de pandemia), en donde se pudo comprobar y medir de forma porcentual que respecto al año anterior a la pandemia (2019), la cantidad de publicaciones

depresivas incrementó en un 31,17% en el año 2020 y se mantuvo casi en la misma cantidad en el año 2021.

7.4. Valoración técnica, económica, ambiental y social

7.4.1. Valoración técnica

El presente TT se valora técnicamente a razón de las múltiples herramientas de hardware y software que fueron utilizadas en la construcción del modelo para identificar depresión. Algunas de las librerías más relevantes usadas son: Twint para la extracción y construcción del dataset con tweets, Scikit Learn para la utilización de los algoritmos de Maquinas de Vectores de Soporte, Random Forest y Naive Bayes, NLTK para realizar un adecuado preprocesamiento de los datos sobre todo en la tokenización y eliminación de stopwords, Stanza para el preprocesamiento en la lematización de los datos, NumPy para realizar cálculos numéricos y almacenamiento de los vectores de características generados y la Librería Imblearn para realizar el sobremuestreo de las clases desequilibradas. Todas estas librerías desempeñaron un papel esencial para lograr el cumplimiento de todas las fases del análisis de sentimientos.

7.4.2. Valoración económica

En el desarrollo del presente TT, fue necesaria la inversión de talento humano, recursos de hardware y software.

Tabla 44. Presupuesto - recursos humanos

Talento Humano	Justificación	N°. horas	Valor por Hora	Subtotal
Autor del proyecto	Estudiante a cargo de la ejecución del proyecto	400	$5	$2000
Personal de apoyo	Docente director en la elaboración y supervisión del proyecto	80	$12.50	$1000
	Docente guía de la materia de Trabajo de Titulación de la Carrera de Ingeniería en Sistemas	40	$12.50	$500

Tabla 45. Presupuesto - recursos HW, SW y TICs

Recursos HW/SW	Justificación	N meses	Valor por mes	Subtotal
Internet	Revisión de literatura y Comunicación.	5	$40	$200
Computadora	Herramienta para el trabajo a desarrollar	5	$10	$50
Mendeley	Gestión de referencias bibliográficas	5	$0	$0
Python	Creación de código y modelos de Machine Learning	5	$0	$0
Jupyter	Creación de código y modelos de Machine Learning	5	$0	$0
Total				**$250**

Tabla 46. Presupuesto total del PTT

Recursos	Subtotal
Recursos Humanos	$3500
Recursos Software. Hardware y TICs	$250
Insumos	$15
Subtotal	$3765
Imprevistos (+10% del Subtotal) *$376,50*	$376.50
Presupuesto total del Proyecto	$4141.50

7.4.3. Valoración ambiental

El presente TT se realizó en su totalidad con recursos tecnológicos y digitales que no tienen un mayor impacto al medio ambiente, además que se tuvo un bajo consumo de recursos materiales o de otros elementos que puedan llegar a perjudicar al medio ambiente.

7.4.4. Valoración social

El presente proyecto ofrece un aporte social ya que se pudo clasificar el sentimiento depresivo en publicaciones de Twitter, con lo cual la información y resultados generados podrían ser usados en el campo de la salud como una herramienta complementaria y de concientización que sirva como un aporte en la mitigación de problemas de salud mental en el Ecuador, además en la educación puede ser una valiosa aportación como una fuente de información para las instituciones educativas, especialmente en colegios y universidades, ya que según menciona la Dra. Ximena Amaya Valarezo en una entrevista (ver anexo 1), en Ecuador nos

hace falta culturizarnos acerca de la importancia de la salud mental. Por lo tanto, esta información puede ayudar a las instituciones a desarrollar nuevas propuestas y metodologías para realizar una intervención preventiva en la salud mental de sus estudiantes.

8. Conclusiones

- El uso de la biblioteca H3 de Uber empleada para generar zonas geográficas del Ecuador dividiendo la zona en conjunto de áreas hexagonales garantizó que se puedan recolectar una gran cantidad de tweets abarcando el mayor territorio posible, lo que permitió que se genere un conjunto de datos adecuado siendo la base para el presente análisis de sentimientos.

- Realizar el entrenamiento de todos los algoritmos con N-gramas permitió conocer el comportamiento de cada uno de estos, comprobándose que todos los modelos que son Maquinas de Vectores de Soporte, Random Forest y Naive Bayes, tienen un mejor rendimiento cuando están entrenados con unigramas, disminuyendo este rendimiento si se entrenan con bigramas y trigramas respectivamente, por lo que se puede confirmar que el texto en los tweets no tienen una estructura estándar en la redacción, evidenciando de esta forma la característica informal en la mayoría de las publicaciones de twitter en Ecuador.

- Comparando los resultados del rendimiento de cada algoritmo de clasificación, se concluye que el algoritmo con mejor resultado en las predicciones es el de Random Forest con una exactitud de 95.4%, sin dejar de lado al algoritmo de Maquinas de Vectores de Soporte que ofrece un rendimiento relativamente cercano con un 95% de exactitud, por lo que también puede ser considerado para clasificar sentimiento depresivo.

- En base a los resultados obtenidos en el análisis univariado y bivariado de los datos se pudo concluir que, las horas en las que se publican más tweets depresivos son en la noche, específicamente entre las 20h00 y 00h00 representando la mayor cantidad de publicaciones, además en base a la cantidad de tweets publicados en el 2020, que es el año cuando empezó el covid-19, se pudo comprobar que la cantidad de tweets depresivos se incrementó en un 31% respecto al año anterior, por lo que en base a

las publicaciones se confirma que la pandemia tuvo un impacto negativo en la salud mental de los ecuatorianos.

9. Recomendaciones

- Se recomienda antes de entrenar los modelos y realizar la detección de sentimiento, resolver el desequilibrio de clases mediante la reproducción de muestras sintéticas de las clases minoritarias, con el objetivo de establecer un equilibrio entre ellas y así aumentar la cantidad de los datos, lo cual es útil cuando se tiene pocos datos en la clase minoritaria, además de evitar que los modelos tengan un sesgo hacia la clase mayoritaria.

- En caso de hacer predicciones con nuevos datos (tweets), se recomienda realizar una limpieza manual para eliminar con mejor eficiencia la información ruidosa, errores gramaticales, abreviaturas o mezcla de distintos lenguajes que son propios de las publicaciones de twitter y que pueden afectar una adecuada clasificación del sentimiento.

- Como trabajo futuro se recomienda utilizar otros tipos de datos para identificar depresión, Por ejemplo, datos biométricos, expresiones faciales del usuario, señales de voz del usuario. Además, se puede implementar utilizando una serie de plataformas de redes sociales adicionales como Facebook e Instagram.

- También como trabajo futuro se recomienda la integración de otras enfermedades mentales como el estrés y el trastorno bipolar, con lo que se podría crear una herramienta más dinámica y flexible.

10. Bibliografía

[1] World Health Organization, “Depression,” 2021. https://www.who.int/news-room/fact-sheets/detail/depression (accessed Dec. 31, 2021).

[2] S. A. Alharthi, “Empirical Study of Features and Unsupervised Sentiment Analysis Techniques for Depression Detection in Social Media,” *Adv. Comput. Sci. Adv Comput Sci*, pp. 3–4, 2020, [Online]. Available: https://www.boffinaccess.com/journals/advances-in-computer-sciences/acs.

[3] S. Zahoor and R. Rohilla, “Twitter Sentiment Analysis Using Machine Learning Algorithms: A Case Study,” in *2020 International Conference on Advances in Computing, Communication Materials (ICACCM)*, 2020, pp. 194–199, doi: 10.1109/ICACCM50413.2020.9213011.

[4] OPS, “Depresión y otros trastornos mentales comunes,” *Organ. Panam. la Salud*, pp. 1–24, 2017, [Online]. Available: http://iris.paho.org/xmlui/bitstream/handle/123456789/34006/PAHONMH17005-spa.pdf.

[5] A. Tusev, L. Tonon, and M. Capella, “Efectos Iniciales en la Salud Mental por la Pandemia de Covid-19 en algunas Provincias de Ecuador,” *Investigatio*, vol. 15, no. 15, pp. 11–22, 2020, doi: 10.31095/investigatio. 2020.15.2.

[6] S. Li, Y. Wang, J. Xue, N. Zhao, and T. Zhu, “The Impact of COVID-19 Epidemic Declaration on Psychological Consequences: A Study on Active Weibo Users,” *Int. J. Environ. Res. Public Health*, vol. 17, no. 6, 2020, doi: 10.3390/ijerph17062032.

[7] OMS, “Plan de acción sobre salud mental 2013-2020,” *Organización Mundial de la Salud*. p. 54, 2013.

[8] World Health Organization, “COVID 19 Public Health Emergency of International Concern (PHEIC) Global Research and Innovation Forum: Towards a Research Roadmap,” *Glob. Res. Collab. Infect. Dis. Prep.*, pp. 1–10, 2020, [Online]. Available: https://www.who.int/publications/m/item/covid-19-public-health-emergency-of-international-concern-(pheic)-global-research-and-innovation-forum.

[9] Confederación Salud Mental España, “Salud mental y COVID-19. Un año de pandemia,” *Confed. Salud Ment. España*, pp. 1–17, 2021.

[10] A. Deshwal and S. K. Sharma, “Twitter Sentiment Analysis using various Classification Algorithms,” in *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, 2016, pp. 251–257, doi: 10.1109/ICRITO.2016.7784960.

[11] B. Liu, *Sentiment Analysis and Opinion Mining*, vol. 5, no. 1. Morgan & Claypool, 2012.

[12] T. Jo, *Text Mining: Concepts, Implementation, and Big Data Challenge*, vol. 26, no. 2. Seoul, Korea (Republic of), 2019.

[13] Z. Rybchak and O. Basystiuk, “Analysis of methods and means of text mining,” *ECONTECHMOD. An Int. Q. J.*, vol. 6, no. 2, pp. 73–78, 2017.

[14] S. A. Salloum, M. Al-Emran, A. A. Monem, and K. Shaalan, “A survey of text mining in social media: Facebook and Twitter perspectives,” *Adv. Sci. Technol. Eng. Syst. J.*, vol. 2, no. 1, pp. 127–133, 2017, doi: 10.25046/aj020115.

[15] M. Chistol and M. Danubianu, “Survey of Text Mining Research Methods and Their Innovative Applicability,” *J. Danubian Stud. Res.*, vol. 11, no. 1, pp. 225–233, 2021.

[16] M. Delgado, M. J. Martín-Bautista, D. Sánchez, and M. A. Vila, “Mining text data: Special features and patterns,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2447, pp. 140–153, 2002, doi: 10.1007/3-540-45728-3_11.

[17] S. A. Salloum, M. Al-Emran, and K. Shaalan, “Mining Text in News Channels: A Case Study from Facebook,” *Int. J. Inf. Technol. Lang. Stud.*, vol. 1, no. 1, pp. 1–9, 2017.

[18] V. Tyag and S. Singh, “Sentiment Analysis to Detect Mental Depression Based on Twitter Data,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 8, no. 9, pp. 268–274, 2020, doi: 10.22214/ijraset.2020.31392.

[19] S. Yang, Z. Ning, and Y. Wu, “NLP Based on Twitter Information: A Survey Report,” *Proc. - 2020 2nd Int. Conf. Inf. Technol. Comput. Appl. ITCA 2020*, pp. 620–625, 2020, doi: 10.1109/ITCA52113.2020.00135.

[20] V. A. Kharde and S. Sonawane, “Sentiment Analysis of Twitter Data : A Survey of Techniques,” *Int. J. Comput. Appl. (0975*, vol. abs/1601.0, 2016, [Online]. Available: http://arxiv.org/abs/1601.06971.

[21] C. Diamantini, A. Mircoli, D. Potena, and E. Storti, “Social information discovery enhanced by sentiment analysis techniques,” *Futur. Gener. Comput. Syst.*, vol. 95, pp. 816–828, 2018, doi: 10.1016/j.future.2018.01.051.

[22] C. Zucco, B. Calabrese, and M. Cannataro, “Sentiment Analysis and Affective Computing for depression monitoring,” *Proc. - 2017 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2017*, vol. 2017-Janua, pp. 1988–1995, 2017, doi: 10.1109/BIBM.2017.8217966.

[23] A. Giachanou and F. Crestani, “Like It or Not: A survey of Twitter Sentiment Analysis Methods,” *ACM Comput. Surv.*, vol. 49, pp. 28:1-28:40, 2016, doi: 10.1145/2938640.

[24] J. F. Raisa, M. Ulfat, A. Al Mueed, and S. M. S. Reza, “A Review on Twitter Sentiment Analysis Approaches,” in *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, 2021, pp. 375–379, doi: 10.1109/ICICT4SD50815.2021.9396915.

[25] F. Azam, M. Agro, M. Sami, M. H. Abro, and A. Dewani, “Identifying Depression among Twitter Users using Sentiment Analysis,” *2021 Int. Conf. Artif. Intell. ICAI 2021*, pp. 44–49, 2021, doi: 10.1109/ICAI52203.2021.9445271.

[26] A. P. Jain and P. Dandannavar, “Application of Machine Learning techniques to Sentiment Analysis,” *Proc. 2016 2nd Int. Conf. Appl. Theor. Comput. Commun. Technol. iCATccT 2016*, pp. 628–632, 2017, doi: 10.1109/ICATCCT.2016.7912076.

[27] D. A. Musleh *et al.*, “Twitter Arabic Sentiment Analysis to detect Depression using Machine Learning,” *Comput. Mater. Contin.*, vol. 71, no. 2, pp. 3463–3477, 2022, doi: 10.32604/cmc.2022.022508.

[28] M. T. Khan, M. Durrani, A. Ali, I. Inayat, S. Khalid, and K. H. Khan, “Sentiment analysis and the complex natural language,” *Complex Adapt. Syst. Model.*, vol. 4, no. 1, 2016, doi: 10.1186/s40294-016-0016-9.

[29] R. Arghandeh *et al.*, “Data Mining Techniques and Tools for Synchrophasor Data,” *North American SynchroPhasor Initiative (NASPI)*. p. 45, 2019, doi: 10.13140/RG.2.2.22389.22242.

[30] K. M. Mendez, L. Pritchard, S. N. Reinke, and D. I. Broadhurst, “Toward collaborative open data science in metabolomics using Jupyter Notebooks and cloud computing,” *Metabolomics*, vol. 15, no. 10, pp. 1–16, 2019, doi: 10.1007/s11306-019-1588-0.

[31] O. Widhyartha, “Engaging People’s Enthusiasm in 2020 Population Census by Scrapping Social Media,” *Expert Meet. Dissem. Commun. Stat.*, no. 1, 2021.

[32] M. Aljabri *et al.*, “Sentiment analysis of Arabic tweets regarding distance learning in Saudi Arabia during the COVID-19 pandemic,” *Sensors*, vol. 21, no. 16, 2021, doi: 10.3390/s21165431.

[33] R. Srinivasan and C. N. Subalalitha, “Sentimental analysis from imbalanced code-mixed data using machine learning approaches,” *Distrib. Parallel Databases*, pp. 1–16, 2021, doi: 10.1007/s10619-021-07331-4.

[34] K. Borowska and J. Stepaniuk, “A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data Gustavo,” *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, 2004, doi: 10.1145/1007730.1007735.

[35] A. Kulkarni, D. Chong, and F. A. Batarseh, “Foundations of data imbalance and solutions for a data democracy,” *CoRR*, pp. 83–106, 2021, doi: 10.48550/arXiv.2108.00071.

[36] A. Vijayvargiya, C. Prakash, R. Kumar, S. Bansal, and J. M. Tavares, “Human knee abnormality detection from Imbalanced sEMG data,” *Biomed. Signal Process. Control*, vol. 66, no. April, pp. 0–35, 2021, doi: 10.1016/j.bspc.2021.102406.

[37] D. M. Aguilera, “Sistema de Detección y Clasificación de Melanomas a través de Imágenes,” Universidad Politécnica de Madrid, 2021.

[38] J. L. Martínez, “Identificación de depresión mediante el análisis de sentimientos,” Universidad de Extremadura, 2019.

[39] Z. Chen and M. Sokolova, “Sentiment Analysis of the COVID-related r/Depression Posts.” arXiv, 2021, doi: 10.48550/ARXIV.2108.06215.

[40] C. Bhargava, S. Poornima, S. Mahur, and M. Pushpalatha, “Depression Detection Using Sentiment Analysis of Tweets,” *Turkish J. Comput. Math. Educ. Res. Artic.*, vol. 12, no. 11, pp. 5411–5418, 2021.

[41] A. Sood, M. Hooda, S. Dhirn, and M. Bhatia, “An Initiative To Identify Depression Using Sentiment Analysis: a Machine Learning Approach,” *Indian J. Sci. Technol.*, vol. 11, no. 4, pp. 1–20, 2018, doi: 10.17485/ijst/2018/v11i4/119594.

[42] Y. Castán, “Introducción al Metodo Cientifico y Sus Etapas,” *Inst. Aragon. Ciencias La Salud*, vol. 2, pp. 1–6, 2006.

[43] J. L. Abreu, “El Método de la Investigación,” *Daena Int. J. Good Conscienc.*, vol. 9, no. 3, pp. 195–204, 2014.

[44] C. A. Espinoza, *Metodología de investigación tecnológica*. 2014.

[45] M. Genero, J. A. Cruz-Lemus, and M. G. Piattini, *Métodos de investigación en ingeniería del software*. 2014.

[46] T. Baviera Puig, “Técnicas para el Análisis de Sentimiento en Twitter: Aprendizaje Automático Supervisado y SentiStrength,” *Dígitos Rev. Comun. Digit.*, vol. 1, no. 3, pp. 33–50, 2017.

[47] A. Saha, A. Al Marouf, and R. Hossain, “Sentiment Analysis from Depression-Related User-Generated Contents from Social Media,” in *2021 8th International Conference on Computer and Communication Engineering (ICCCE)*, 2021, pp. 259–264, doi: 10.1109/ICCCE50029.2021.9467214.

[48] M. I. Sajib, S. Mahmud Shargo, and M. A. Hossain, “Comparison of the efficiency of

Machine Learning algorithms on Twitter Sentiment Analysis of Pathao," in *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, 2019, pp. 1–6, doi: 10.1109/ICCIT48885.2019.9038208.

[49] S. Tiwari, A. Verma, P. Garg, and D. Bansal, "Social Media Sentiment Analysis On Twitter Datasets," in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 2020, pp. 925–927, doi: 10.1109/ICACCS48705.2020.9074208.

[50] M. Deshpande and V. Rao, "Depression detection using emotion artificial intelligence," *Proc. Int. Conf. Intell. Sustain. Syst. ICISS 2017*, no. Iciss, pp. 858–862, 2018, doi: 10.1109/ISS1.2017.8389299.

[51] J. A. Mansilla, "Minado de texto aplicado en Twitter para obtener la polaridad en opiniones de usuarios acerca den nuevo Proyecto de ley que regula la migración en Chile," Universidad del Bío-Bío, 2018.

[52] L. Liu, *Encyclopedia of Database Systems*, 1st ed. Springer, 2009.

[53] W. Elmenreich, J. T. Machado, and I. J. Rudas, *Intelligent Systems at the Service of the Mankind, vol. 1*. 2003.

[54] S. M. Fonseca *et al.*, "An approach based on text mining for knowledge acquisition in diagnostic systems," 2007.

[55] S. J. Pachouly, G. Raut, K. Bute, R. Tambe, and S. Bhavsar, "Depression Detection on Social Media Network (Twitter) using Sentiment Analysis," *Int. Res. J. Eng. Technol.*, vol. 08, no. 01, 2021, [Online]. Available: www.irjet.net.

[56] B. A. Eclarin, A. C. Fajardo, and R. P. Medina, "A novel feature hashing with efficient collision resolution for bag-of-words representation of text data," *ACM Int. Conf. Proceeding Ser.*, pp. 12–16, 2018, doi: 10.1145/3278293.3278301.

[57] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Syst. Appl.*, vol. 57, pp. 117–126, 2016, doi: 10.1016/j.eswa.2016.03.028.

[58] N. V. Babu and E. G. M. Kanaga, "Sentiment Analysis in Social Media Data for Depression Detection Using Artificial Intelligence: A Review," *SN Comput. Sci.*, vol. 3, no. 74, pp. 1–20, 2021, doi: 10.1007/s42979-021-00958-1.

[59] A. P. Tirtopangarsa and W. Maharani, "Sentiment Analysis of Depression Detection on Twitter Social Media Users Using the K-Nearest Neighbor Method," *Inov. Teknol. dan Pengolah. Inf. untuk Mendukung Transform. Digit.*, vol. 1, no. 1, pp. 247–258, 2021.

[60] W. McKinney, *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. 2017.

[61] B. Bondaruk, S. A. Roberts, and C. Robertson, "Assessing the state of the art in Discrete Global Grid Systems: OGC criteria and present functionality," *Geomatica*, vol. 74, no. 1, pp. 9–30, 2020, doi: 10.1139/geomat-2019-0015.

[62] I. Lauriola, A. Lavelli, and F. Aiolli, "An introduction to Deep Learning in Natural Language Processing: Models, techniques, and tools," *Neurocomputing*, vol. 470, pp. 443–456, 2021, doi: 10.1016/j.neucom.2021.05.103.

[63] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, "Stanza: A Python Natural Language Processing Toolkit for Many Human Languages," *CoRR*, pp. 101–108, 2020, doi: 10.18653/v1/2020.acl-demos.14.

[64] L. Sohmen and L. Rossenova, “Open refine to wikibase: a new data upload pipeline,” in *Proceedings of the 22nd ACM/IEEE Joint Conference on Digital Libraries*, 2022, no. 53, pp. 1–2, doi: 10.1145/3529372.3530919.

[65] O. Kramer, “Scikit-Learn,” in *Machine Learning for Evolution Strategies*, Springer International Publishing, 2016, pp. 45–53.

[66] G. Zhao, Y. Liu, W. Zhang, and Y. Wang, “TFIDF based feature words extraction and topic modeling for short text,” *ACM Int. Conf. Proceeding Ser.*, pp. 188–191, 2018, doi: 10.1145/3180374.3181354.

[67] D. Sarkar, “Feature Engineering for Text Representation,” in *Text Analytics with Python: A Practitioner’s Guide to Natural Language Processing*, Apress, 2019, pp. 201–273.

[68] J. Brownlee, *XGBoost With Python: Gradient Boosted Trees with XGBoost an scikit-learn*, vol. v1.10. 2018.

[69] P. Singh and A. Manure, *Learn TensorFlow 2.0: Implement Machine Learning and Deep Learning Models with Python*. Bangalore: Apress, 2020.

[70] U. Saeed, S. Ullah Jan, Y. D. Lee, and I. Koo, “Machine Learning-based Real-Time Sensor Drift Fault Detection using Raspberry Pi,” *2020 Int. Conf. Electron. Information, Commun. ICEIC 2020*, vol. 2020-Janua, pp. 4–10, 2020, doi: 10.1109/ICEIC49074.2020.9102342.

[71] A. Leis, F. Ronzano, M. A. Mayer, L. I. Furlong, and F. Sanz, “Detecting signs of depression in tweets in Spanish: Behavioral and linguistic analysis,” *J. Med. Internet Res.*, vol. 21, no. 6, 2019, doi: 10.2196/14199.

[72] M. Park, C. Cha, and M. Cha, “Depressive moods of users portrayed in Twitter,” *Proc. 18th ACM Int. Conf. Knowl. Discov. Data Mining, SIGKDD 2012*, pp. 1–8, 2012.

[73] M. Choudhury, M. Gamon, S. Counts, and E. Horvitz, “Predicting Depression via Social Media,” in *Proceedings of the International AAAI Conference on Webolgs and Social Media*, 2013, vol. 7, no. 1, pp. 128–137, doi: 10.3109/01460862.2013.798190.

[74] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, “How many trees in a random forest?,” in *Machine Learning and Data Mining in Pattern Recognition*, no. July 2012, Springer Berlin Heidelberg, 2012, pp. 154–168.

[75] Servicio Nacional de Gestión de Riesgo y Emergencias, “Informe de situación COVID-19 Ecuador 16 de Marzo de 2020,” 2020. [Online]. Available: https://www.gestionderiesgos.gob.ec/wp-content/uploads/2020/03/Informe-de-Situación-No008-Casos-Coronavirus-Ecuador-16032020-20h00.pdf.

[76] M. Ebner, “Microblogs,” *SAGE Encycl. Internet*, pp. 640–641, 2018, doi: 10.4135/9781473960367.n183.

[77] J. Gutiérrez-Rexach, *ENCICLOPEDIA DE LINGÜÍSTICA HISPÁNICA*, 1st ed. Routledge, 2016.

11. Anexos

11.1. Anexo 1

Entrevista realizada con el propósito de sustentar el presente TT, tanto a nivel social como académico. Como apoyo de la información captada en la entrevista, se adjunta la grabación de las misma en el siguiente enlace:

https://drive.google.com/drive/folders/1XNtTVpQkyxXpBouewN1hgeLNijI6cflq?usp=sharing

A continuación, se redacta fielmente las respuestas vertidas por su autor.

Cargo: Psicóloga Clínica

Nombre: Ximena Dennisse Amaya Valarezo

Fecha de entrevista: 21-01-2022

Descripción:

1. **¿Considera a la depresión como un problema de salud serio?**

Claro que sí. De hecho, dentro de los nueve años que tengo ejerciendo la carrera es uno de los trastornos que más se trabaja dentro de sesión. Lo dice la Organización Mundial de la Salud, que dentro de la población en general, en los adultos, dentro de toda la población hay un **5%** de personas que lo padecen. Entonces es una enfermedad que lleva a tener bastantes alteraciones psicológicas, inclusive lleva a cometer el acto de suicidarse. Por eso es muy importante detectar a tiempo los síntomas para poder trabajar con los pacientes que padezcan episodios depresivos o que conlleven a un trastorno depresivo mayor o crónico que tenga como finalidad la muerte. Creo que es una de las enfermedades con las que más se batalla últimamente y que en los últimos tiempos se ha logrado detectar con esta situación de la pandemia.

2. ¿Cree que las medidas de restricción tomadas como consecuencia del covid ha repercutido negativamente sobre la salud mental de las personas en Ecuador?

En cierta parte, creo que el Ecuador, o bueno, quizá algunos países también como Colombia, como Cuba, como Puerto Rico, son países que creo que les hace falta bastante culturizarnos con el tema de la prevención y de la convivencia. Y eso nos jugó un papel importante en la pandemia en el 2020, porque hubo muchos hogares en donde les afectó bastante el tema de la convivencia. Los divorcios aumentaron muchísimo y lógicamente que aumentó en el tema de la convivencia y para las personas que padecieron la enfermedad, ahí hubo casos de depresión, de ansiedad y luego estrés postraumático. Personas que también no tuvieron la

enfermedad, pero sí tuvieron casos de tuvieron episodios de ansiedad y de ataques de pánico porque tenían miedo de padecer lo que es el virus. Ahora la situación ha cambiado un poquito porque quizás antes como no habían vacunas para la enfermedad, era como que más latente y los síntomas eran más fuertes y hacía que nosotros nos enfermemos más.

Pero tiene que ver mucho la parte mental, porque sí nosotros los seres humanos no batallamos o no trabajamos con nuestros pensamientos. Es lo que nos lleva a tener estos ataques de pánico y ansiedad. Y los pensamientos recurrentes en el 2020 eran sobre ¿Y si me enfermo? ¿Y si contagio en mi familia? ¿Y si me llego a morir? Eran ya pensamientos catastróficos. Ahora, ya que existen las vacunas, quizás hay personas que se han contagiado pero los síntomas son un poquito más leves y como que quizá podría ser que nos hemos acostumbrado un poquito a la pandemia, pero aún hace falta trabajar en el tema de la prevención y de la cultura. Creo que sí ha afectado en las familias. El año pasado atendí bastantes casos de divorcio y bastantes casos en donde los pacientes somatizaban todo lo que sentían, el estrés, la ansiedad que tenían dentro del día. Lo somatizaban con enfermedades como insomnio, como problemas gastrointestinales, como dolores de cabeza recurrentes que provenían del estrés y de la ansiedad que no fue tratada a tiempo. Por eso es muy importante acudir al psicólogo.

3. ¿Existe estigma o renuencia asociado con buscar ayuda profesional para los problemas mentales, incluyendo la depresión?

Hablando de 100% de población en cuanto a Ecuador, yo diría que el 40%, por no decir el 35%, todavía considera que ir al psicólogo es para los “locos”. No tomamos en cuenta de que, así como nuestra salud física es muy importante también la salud mental. De qué nos sirve trabajar en nuestro cuerpo, en nuestra alimentación, en nuestro autocontrol, si no cuidamos lo que tenemos en nuestra mente. Nosotros, los seres humanos en el día procesamos entre dos mil a tres mil pensamientos al día. Ahora hay que ver si en las personas, esos pensamientos son más negativos o más positivos que lo que nos decimos a nosotros mismos. Entonces es tan importante ir al psicólogo, no solamente para cuando tenemos algún problema, sino también para saber cómo estamos llevando nuestra vida. Entonces, hoy por hoy, si hay personas que consideran de que ir al psicólogo es solamente cuando es algo está mal y todavía hace falta culturizarnos un poquito más acerca de la importancia del bienestar mental.

4. ¿Considera que las redes sociales como Twitter pueden ser un medio para que las personas que sufren depresión puedan expresar sus sentimientos y opiniones?

Como profesional de la salud mental, siempre yo voy a sugerir acudir a un terapeuta, porque no es lo mismo escuchar un consejo que te dicen en internet o que te lo dice la vecina, o que te lo dice tu mamá, tu papá o algún hermano o hermana, a que te lo diga un profesional que con bases y herramientas psicológicas te puede atender y te puede ayudar a buscar una solución correcta ante los problemas que tú tienes. Entonces el tema de las redes sociales es un tema complejo porque hay un pro y un contra. Quizá el pro es que nosotros podemos utilizar las redes sociales cuando tenemos mucha inteligencia emocional, ¿A qué me refiero?, a que podemos tener un equilibrio entre qué es lo bueno y qué es lo malo y tener una madurez como para poder receptar los comentarios y las opiniones ajenas a lo que nosotros sentimos. Ahora, ¿cuál es el contra?, cuando personas que tienen alguna alteración psicológica como, por ejemplo, están atravesando alguna crisis de ansiedad, o están pasando por un cuadro psicológico o depresivo, o quizá tienen depresión, pero no se han dado cuenta, no tienen esa sabiduría o esa inteligencia emocional para poder proyectar o para poder percibir un comentario, ¿a qué me refiero?, hay muchas personas que tienen depresión, arrastran episodios depresivos en ciclos de su vida, pero como no lo han detectado tiempo, no han sido tratados. Entonces eso conlleva que tomen malas decisiones, a que tengan intolerancia a la frustración, a que se enfermen con mayor facilidad y no puedan gestionar sus emociones.

Entonces, si un depresivo crónico, quizá en su en medio de su alteración, lee dentro de las redes sociales la vida no sirve de nada, de que es una persona insensible o lo atacan con comentarios negativos, quizá no tenga el suficiente valor o la suficiente madurez como para poder aceptar ese comentario. Y puede jugar un papel importante como para que tome la decisión de quitarse la vida o tome la decisión que los otros están haciendo, cuando en realidad lo importante es poder tener un equilibrio y poder gestionar nuestras emociones de la forma más adecuada. Creo que las redes sociales nos ayudarían, siempre y cuando nosotros tengamos una madurez emocional equilibrada. Si nosotros no tenemos la madurez emocional que se requiere, como aceptar la opinión de la otra persona sin que me afecte a mí, o leer un comentario brusco hacia mi persona pero que no me llegue a afectar como para atentar contra mi vida o bajar mis niveles de autoestima creo que está bien, pero si me afecta, me duele y me va a conllevar a que yo tenga ataques de ira, a que me desquite con la persona que está al lado, a que tome decisiones fatales en mi vida. Yo creo que no estoy preparada para usar las redes sociales, entonces las redes sociales también incluyen el hecho de que nosotros podamos trabajarlo en familia. Hay muchos hogares, padres de familia, en donde no hay la supervisión de las redes sociales en los teléfonos de los chicos.

Hablando de que de que un niño de 8 años en adelante debe tener supervisión para tener un teléfono y tener una red social porque no tiene la madurez necesaria. Y ahora en las redes sociales hay bastantes plataformas sexuales, bastantes plataformas en donde conllevan a

que los niños hagan daño o que haya bastante. Hay bastantes actos de violencia y los niños cuando no tienen una buena estructura mental porque hay deficiencia en su hogar como problemas, conflictos, adicciones, conllevan a que no tengan una estabilidad emocional y una buena gestión de sus emociones y que cometan actos atroces como matar, hacerse daño. Hace como dos años hubo una situación de un juego virtual que hacían con una ballena, de que mientras dibujaban una ballena ellos tenían que dibujarse la ballena en el brazo y se iban cortando el brazo, pero ellos no se daban cuenta de que se estaban haciendo daño e inclusive podían provocarse la muerte. Entonces yo creo que el tema de las redes sociales es un tema bastante complejo. Tenemos que llevar mucha supervisión en el caso de los niños y adolescentes y creo que podría ser conllevado a que a que tengamos una buena familiarización con las redes sociales, siempre y cuando nosotros tengamos. Inteligencia emocional, y sepamos qué es lo bueno, que es lo malo y que podamos aceptar las opiniones del resto.

11.2. Anexo 2

Encuesta

Cargo: PSICOLOGO DE LA UNIDAD DE BIENESTAR UNIVERSITARIO
Nombre: JORGE FERNANDO JIMENEZ SANCHEZ
Fecha de encuesta: 17 DE ENERO DE 2022

Descripción:

1. ¿Considera a la depresión como un problema de salud serio?

La depresión es causada por una combinación de factores genéticos, biológicos, ambientales y psicológicos. Por lo tanto, es un serio problema de salud.

2. ¿Cree que las medidas de restricción tomadas como consecuencia del covid ha repercutido negativamente sobre la salud mental de las personas en Ecuador?

Si. Por no haber un plan de contingencia adecuado, en donde la parte psicológica acompañe a la médica para poder reducir el número de personas deprimidas, ansiosas, nerviosas, acompañadas de llanto etc. Apoyados con planes terapéuticos.

3. ¿Existe estigma o renuencia asociado con buscar ayuda profesional para los problemas mentales, incluyendo la depresión?

Si existe, debido a que todo el mundo piensa que no necesita de ayuda profesional y por ende en estos casos las personas se guardan sus problemas emocionales lo que en poco tiempo se agravan hasta llegar a la depresión.

4. ¿Considera que las redes sociales como Twitter pueden ser un medio para que las personas que sufren depresión puedan expresar sus sentimientos y opiniones?

Las personas depresivas tienen un lenguaje más pobre: sienten menos interés por lo que les rodea y eso hace que no usen frases enteras para expresarse.

También son personas que retuitean menos, interactúan menos con su entorno, y que usan palabras con polaridad negativa.

11.3. Anexo 3

Geodatos generados para el territorio de Ecuador.

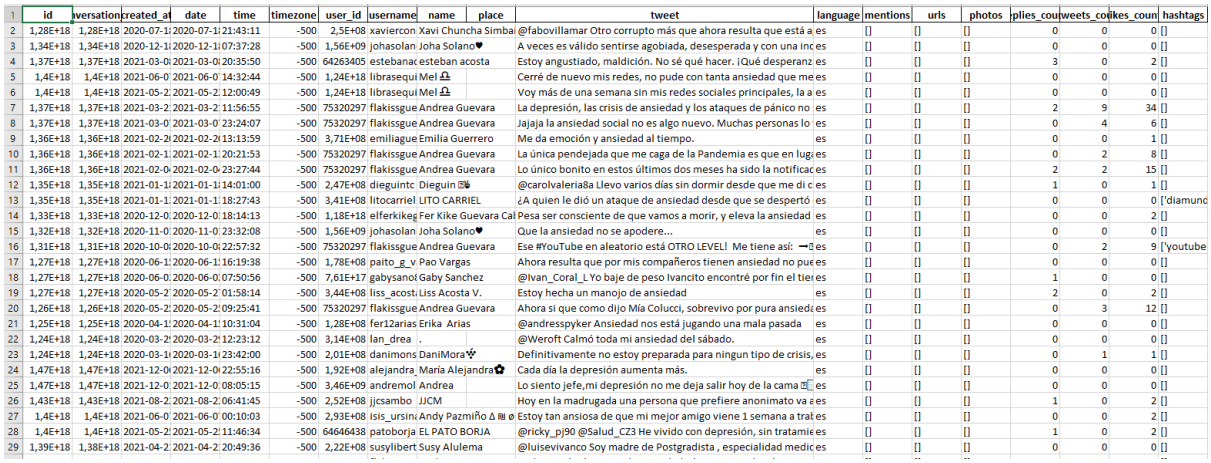
Figura 46. Geodatos de Ecuador obtenidos mediante la herramienta geoson.io

Para ver el archivo Geojson que se generó del territorio de Ecuador que se puede ver en la figura #, está disponible en esta dirección:

https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/Extraccion_Tweets/area_ecuador.geojson

11.4. Anexo 4

- Captura de una previa del dataset de tweets depresivos:



	id	versation	created_at	date	time	timezone	user_id	username	name	place	tweet	language	mentions	urls	photos	replies_cou	retweets_co	likes_coun	hashtags
2	1,28E+18	1,28E+18	2020-07-1	2020-07-1	21:43:11	-500	2,5E+08	xaviercon	Xavi Chuncha Simba		@fabovillamar Otro corrupto más que ahora resulta que está a	es	[]	[]	[]	0	0	0	[]
3	1,34E+18	1,34E+18	2020-12-1	2020-12-1	07:37:28	-500	1,56E+09	johasolan	Joha Solano♥		A veces es válido sentirse agobiada, desesperada y con una inc	es	[]	[]	[]	0	0	0	[]
4	1,37E+18	1,37E+18	2021-03-0	2021-03-0	20:35:50	-500	64263405	estebanac	esteban acosta		Estoy angustiado, maldición. No sé qué hacer. ¡Qué desperanz	es	[]	[]	[]	3	0	2	[]
5	1,4E+18	1,4E+18	2021-06-0	2021-06-0	14:32:44	-500	1,24E+18	librasequi	Mel ♎		Cerré de nuevo mis redes, no pude con tanta ansiedad que me	es	[]	[]	[]	0	0	0	[]
6	1,4E+18	1,4E+18	2021-05-2	2021-05-2	12:00:49	-500	1,24E+18	librasequi	Mel ♎		Voy más de una semana sin mis redes sociales principales, la a	es	[]	[]	[]	0	0	0	[]
7	1,37E+18	1,37E+18	2021-03-2	2021-03-2	11:56:55	-500	75320297	flakissgue	Andrea Guevara		La depresión, las crisis de ansiedad y los ataques de pánico no	es	[]	[]	[]	2	9	34	[]
8	1,37E+18	1,37E+18	2021-03-0	2021-03-0	23:24:07	-500	75320297	flakissgue	Andrea Guevara		Jajaja la ansiedad social no es algo nuevo. Muchas personas lo	es	[]	[]	[]	0	4	6	[]
9	1,36E+18	1,36E+18	2021-02-2	2021-02-2	13:13:59	-500	3,71E+08	emiliague	Emilia Guerrero		Me da emoción y ansiedad al tiempo.	es	[]	[]	[]	0	0	1	[]
10	1,36E+18	1,36E+18	2021-02-1	2021-02-1	20:21:53	-500	75320297	flakissgue	Andrea Guevara		La única pendejada que me caga de la Pandemia es que en lug	es	[]	[]	[]	0	2	8	[]
11	1,36E+18	1,36E+18	2021-02-0	2021-02-0	23:27:44	-500	75320297	flakissgue	Andrea Guevara		Lo único bonito en estos últimos dos meses ha sido la notificac	es	[]	[]	[]	2	2	15	[]
12	1,35E+18	1,35E+18	2021-01-1	2021-01-1	14:01:00	-500	2,47E+08	dieguintc	Dieguin 🇪🇨		@carolvaleria8a Llevo varios días sin dormir desde que me di c	es	[]	[]	[]	1	0	1	[]
13	1,35E+18	1,35E+18	2021-01-1	2021-01-1	18:27:43	-500	3,41E+08	litocarriel	LITO CARRIEL		¿A quien le dió un ataque de ansiedad desde que se despertó	es	[]	[]	[]	0	0	0	['diamund
14	1,33E+18	1,33E+18	2020-12-0	2020-12-0	18:14:13	-500	1,18E+18	elferkikeg	Fer Kike Guevara Cal		Pesa ser consciente de que vamos a morir, y eleva la ansiedad	es	[]	[]	[]	0	0	2	[]
15	1,32E+18	1,32E+18	2020-11-0	2020-11-0	23:32:08	-500	1,56E+09	johasolan	Joha Solano♥		Que la ansiedad no se apodere...	es	[]	[]	[]	0	0	0	[]
16	1,31E+18	1,31E+18	2020-10-0	2020-10-0	22:57:32	-500	75320297	flakissgue	Andrea Guevara		Ese #YouTube en aleatorio está OTRO LEVEL! Me tiene así: 🤯	es	[]	[]	[]	0	2	9	['youtube
17	1,27E+18	1,27E+18	2020-06-1	2020-06-1	16:19:38	-500	1,78E+08	paito_g_v	Pao Vargas		Ahora resulta que por mis compañeros tienen ansiedad no pue	es	[]	[]	[]	0	0	0	[]
18	1,27E+18	1,27E+18	2020-06-0	2020-06-0	07:50:56	-500	7,61E+17	gabysanc	Gaby Sanchez		@Ivan_Coral_L Yo baje de peso Ivancito encontré por fin el tie	es	[]	[]	[]	1	0	0	[]
19	1,27E+18	1,27E+18	2020-05-2	2020-05-2	01:58:14	-500	3,44E+08	liss_acost	Liss Acosta V.		Estoy hecha un manojo de ansiedad	es	[]	[]	[]	2	0	2	[]
20	1,26E+18	1,26E+18	2020-05-2	2020-05-2	09:25:41	-500	75320297	flakissgue	Andrea Guevara		Ahora si que como dijo Mía Colucci, sobrevivo por pura ansied	es	[]	[]	[]	0	3	12	[]
21	1,25E+18	1,25E+18	2020-04-1	2020-04-1	10:31:04	-500	1,28E+08	fer12arias	Erika Arias		@andresspyker Ansiedad nos está jugando una mala pasada	es	[]	[]	[]	0	0	0	[]
22	1,24E+18	1,24E+18	2020-03-2	2020-03-2	12:23:12	-500	3,14E+08	lan_drea	.		@Weroft Calmó toda mi ansiedad del sábado.	es	[]	[]	[]	0	0	0	[]
23	1,24E+18	1,24E+18	2020-03-1	2020-03-1	23:42:00	-500	2,01E+08	danimons	DaniMora🐈‍⬛		Definitivamente no estoy preparada para ningun tipo de crisis,	es	[]	[]	[]	0	1	1	[]
24	1,47E+18	1,47E+18	2021-12-0	2021-12-0	22:55:16	-500	1,92E+08	alejandra	María Alejandra✿		Cada día la depresión aumenta más.	es	[]	[]	[]	0	0	0	[]
25	1,47E+18	1,47E+18	2021-12-0	2021-12-0	08:05:15	-500	3,46E+09	andremol	Andrea		Lo siento jefe,mi depresión no me deja salir hoy de la cama 🛌	es	[]	[]	[]	0	0	0	[]
26	1,43E+18	1,43E+18	2021-08-2	2021-08-2	06:41:45	-500	2,52E+08	jjcsambo	JJCM		Hoy en la madrugada una persona que prefiere anonimato va a	es	[]	[]	[]	1	0	2	[]
27	1,4E+18	1,4E+18	2021-06-0	2021-06-0	00:10:03	-500	2,93E+08	isis_ursin	Andy Pazmiño Δ 🇪🇨 ø		Estoy tan ansiosa de que mi mejor amigo viene 1 semana a trat	es	[]	[]	[]	0	0	2	[]
28	1,4E+18	1,4E+18	2021-05-2	2021-05-2	11:46:34	-500	64646438	patoborja	EL PATO BORJA		@ricky_pj90 @Salud_CZ3 He vivido con depresión, sin tratamie	es	[]	[]	[]	1	0	2	[]
29	1,39E+18	1,38E+18	2021-04-2	2021-04-2	20:49:36	-500	2,22E+08	susylibert	Susy Alulema		@luisevivanco Soy madre de Postgradista , especialidad medic	es	[]	[]	[]	0	0	0	[]

Figura 47. Captura parcial del dataset inicial de los tweets recopilados relacionados a depresión.

Para acceder al archivo completo con los datos iniciales que se muestran en la figura 7, está disponible en la siguiente dirección:

https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/Extraccion_Tweets/Dataset_depresivo_inicial.xlsx

- Captura de una previa del dataset de tweets aleatorios:

	id	versation	created_at	date	time	timezone	user_id	username	name	place	tweet	language	mentions	urls	photos	replies_cou	retweets_co	likes_c
2	1,48E+18	1,48E+18	2021-12-2	2021-12-2	18:59:27	-500	1,1E+18	eddyshing	Eddy René Shingre M		@maripydev No es necesario pero si ayuda en mucho. En mi caso yo soy Matem	es	[]	[]	[]	0	0	
3	1,48E+18	1,48E+18	2021-12-2	2021-12-2	18:51:45	-500	4,73E+08	santiagoc	santiago chavez		@muymariana_ec Soy mal trabajador de antemano jajaja	es	[]	[]	[]	0	0	
4	1,48E+18	1,48E+18	2021-12-2	2021-12-2	17:40:23	-500	2,86E+09	garfjenny	J.B ♐ ♑		por eso no me juzgues si no te gusta algo de mí, yo soy como soy, irreverente, e	es	[]	[]	[]	0	0	
5	1,48E+18	1,48E+18	2021-12-2	2021-12-2	17:37:06	-500	2,86E+09	garfjenny	J.B ♐ ♑		También soy delicada y dulce, soy ángel y demonio, soy una demente complet	es	[]	[]	[]	1	0	
6	1,48E+18	1,48E+18	2021-12-2	2021-12-2	17:34:48	-500	2,86E+09	garfjenny	J.B ♐ ♑		si estoy mal y tú también; no dudes que correré a levantarte, porque así soy yo.	es	[]	[]	[]	1	0	
7	1,48E+18	1,48E+18	2021-12-2	2021-12-2	16:58:28	-500	1,07E+08	conehex7	Conejito 🖤		@QUIBARINT @jfloresluna @estebanavila Tu : la tiza es igual que la tusa que in	es	[]	[]	[]	1	0	
8	1,48E+18	1,48E+18	2021-12-2	2021-12-2	16:44:53	-500	1,18E+18	besaletty	Letty BeSa		Soy todo lo que quieres, pero al revés.	es	[]	[]	[]	0	1	
9	1,48E+18	1,48E+18	2021-12-2	2021-12-2	16:03:46	-500	2,3E+08	juanchito_	Juan Carlos Pacheco.		Bueno mijo se me acabó el fin de año!! Gripe hpt!!	es	[]	[]	[]	0	0	
10	1,48E+18	1,48E+18	2021-12-2	2021-12-2	14:56:52	-500	69426546	fernandol	*Fernando Lippke*		Por otro lado: todos se quejan de los impuestos (yo soy uno de ellos) y la chire	es	[{'screen_	[]	[]	0	0	
11	1,48E+18	1,48E+18	2021-12-2	2021-12-2	13:52:12	-500	5,8E+08	macias_os	Oswaldo Macias		@jefrig105 Claro, y para mi el Vallenato es el mejor ritmo autóctono que existe	es	[]	[]	[]	0	0	
12	1,48E+18	1,48E+18	2021-12-2	2021-12-2	13:37:11	-500	1,45E+18	ndkainapa	Kaina Pachay		Merezco lo mejor del mundo porque no soy mala persona y ya aguanté mucha r	es	[]	[]	[]	0	0	
13	1,48E+18	1,48E+18	2021-12-2	2021-12-2	13:20:18	-500	3,43E+08	pamecavi	Pame Caviedes		@stalinvistinc Jajajajajaja si soy 🤦	es	[]	[]	[]	1	0	
14	1,48E+18	1,48E+18	2021-12-2	2021-12-2	12:21:30	-500	37415600	manenau	Manena Uzcátegui E		@guillen_blanchy @julio_efe_ Primero: NO soy antivacuna. Segundo: NO le pu	es	[]	[]	[]	1	0	
15	1,48E+18	1,48E+18	2021-12-2	2021-12-2	12:03:29	-500	1,18E+18	od3ray	Oderay Andrade		Soy sentimental y lloro al final de las películas, y muchas veces durante una dis	es	[]	[]	[]	0	0	
16	1,48E+18	1,48E+18	2021-12-2	2021-12-2	11:54:07	-500	2,82E+08	ivanbuitrc	Ivan Buitron		@soy_502 @MinSaludGuate 🇬🇹🇬🇹	und	[]	[]	[]	0	0	
17	1,48E+18	1,48E+18	2021-12-2	2021-12-2	11:03:49	-500	2,59E+09	lslucerom	Stheffy Lucero		Estoy haciendo limpieza de mi ropero y hasta mis maquillajes y putaaa hay pren	es	[]	[]	[]	0	0	
18	1,48E+18	1,48E+18	2021-12-2	2021-12-2	10:34:59	-500	4,01E+08	peteranrr	LemonKiki		Indirectas? Tiros por elevación? Yo ya soy un hombre grande para esas huevadi	es	[]	[]	[]	0	0	
19	1,48E+18	1,48E+18	2021-12-2	2021-12-2	10:16:20	-500	2,44E+08	elgranmja	MARCOS JARA GARC		@CantaditoCUE @tabatalarab @pedropalaciosu @HUGOEPALACIOSU1 @tomet	es	[]	[]	[]	1	0	
20	1,48E+18	1,48E+18	2021-12-2	2021-12-2	09:54:36	-500	9,44E+17	teestoymi	TEESTOYMIRANDO		@JazminM58648273 @FaustoJarrinT NO SOY DE LASSO, PEOR DE CORREA, SOY E	es	[]	[]	[]	2	0	
21	1,48E+18	1,48E+18	2021-12-2	2021-12-2	09:28:18	-500	4,95E+08	daniel982	Daniel Isaac		Soy yo o todo el mundo quiere terminar borracho en fin de año.	es	[]	[]	[]	2	0	
22	1,48E+18	1,48E+18	2021-12-2	2021-12-2	08:54:37	-500	1,52E+09	galarzalay	Layla Galarza		@luisevivanco Que pena, soy solidaria!	es	[]	[]	[]	0	0	
23	1,48E+18	1,48E+18	2021-12-2	2021-12-2	08:39:27	-500	1,39E+08	lissetteug	Lissette Ugalde		@payaguilargame Soy la burla del Instagram gracias al personaje asignado el día	es	[]	[]	[]	1	0	
24	1,48E+18	1,48E+18	2021-12-2	2021-12-2	07:57:30	-500	4,05E+09	nlmoraf	Leonel mora		@GiovanniTrave51 @verosoficoronel Haaa ,yo soy correista, jajajajajajaja	es	[]	[]	[]	0	0	
25	1,48E+18	1,48E+18	2021-12-2	2021-12-2	01:45:28	-500	1,91E+08	cristhian_	Aaron		Soy de las personas que a esta hora ven tik toks de gente fileteando un pescad	es	[]	[]	[]	0	0	
26	1,48E+18	1,48E+18	2021-12-2	2021-12-2	01:23:02	-500	9,96E+17	soycmji	Claudia 🇪🇨		Hola Dios soy yo de nuevo. Una semana de reto puro. Pero no dejaremos que n	es	[]	[]	[]	1	0	
27	1,48E+18	1,48E+18	2021-12-2	2021-12-2	01:17:04	-500	1,94E+08	williamga	William Garzon		Que si soy bendecido ? Marica gano plata por hacer lo que amo 🤑	es	[]	[]	[]	1	1	
28	1,48E+18	1,48E+18	2021-12-2	2021-12-2	01:12:38	-500	1,56E+09	paolo_vel	Paolo 🇪🇨		Soy inseguro	es	[]	[]	[]	0	0	
29	1,48E+18	1,48E+18	2021-12-2	2021-12-2	00:28:18	-500	4,36E+08	juanpanch	Juanfranciscopillajo		@OrlandoPerezEC @CorteNacional @CorteConstEcu será q la mayoría es odio	es	[]	[]	[]	0	0	
30	1,48E+18	1,48E+18	2021-12-2	2021-12-2	23:35:33	-500	1,46E+18	rafitapuni	@tuRomanticoPúblic		@carrasquerorest yo soy una persona muy simpática y conquistador🔥	es	[]	[]	[]	0	0	

Figura 48. Captura parcial del dataset inicial de los tweets aleatorios recopilados.

Para acceder al archivo completo con los datos iniciales que se muestran en la figura 8, está disponible en la siguiente dirección:

https://github.com/byronmb/Identificacion_Depresion_Ecuador/blob/main/Extraccion_Tweets/Dataset_random_inicial.xlsx