Interfaz gráfica para análisis y clasificación de sintomatología asociada a la COVID-19 utilizando *machine learning*

Graphical Interface for Analysis and Classification of Symptomatology associated with COVID-19 using Machine Learning

Soto-Niño, Brayner Sneyder[1], Ramirez-Bautista, Julian Andres[2]

Fundación Universitaria de San Gil, UNISANGIL
Facultad de Ciencias Naturales e Ingeniería
San Gil, Colombia

braynersoto@unisangil.edu.co
jramirez@unisangil.edu.co

Fecha de recepción: noviembre 22 de 2022
Fecha de aceptación: junio 29 de 2023

Resumen **—** La COVID-19 planteó grandes retos a la comunidad científica, la cual, además de centrar esfuerzos en el desarrollo de una vacuna, dio espacio para la formulación de estrategias que permitiesen controlar la propagación del virus en lo que se conseguía una inmunidad efectiva. En este escenario, la inteligencia artificial se convirtió en una herramienta aprovechable para el análisis de la sintomatología preliminar. En el trabajo se muestra el desarrollo de un sistema basado en algoritmos de aprendizaje automático para la identificación de pacientes COVID-19, para lo cual se utilizó una base de datos de la plataforma *Kaggle* (Comunidad de Científicos de datos y profesionales del *Machine Learning*) con registro de 5.434 personas, de las cuales 1.051 eran casos negativos y 4.383 casos positivos para la enfermedad. El trabajo determinó los síntomas más relevantes, y definió un algoritmo de aprendizaje automático para la clasificación y análisis de datos, el cual arrojó como métrica de desempeño un puntaje F1 de 0.98 y una *aucROC* de 0.99. Se desarrolló un entorno gráfico para integrar el modelo, utilizando *Jupyter*, *QtDesinger* y *PyQt5*, para dar facilidad al usuario en el manejo del modelo de red neuronal artificial obtenida.

Palabras clave— Sintomatología COVID, prevención, datos fisiológicos, RNA, indicadores de desempeño, entropía binaria cruzada.

Abstract **-** COVID-19 posed great challenges to the scientific community, which, in addition to focusing efforts on the development of a vaccine, gave space for the formulation of strategies to control the spread of the virus while effective immunity was being achieved. In this scenario, artificial intelligence became a useful tool for the analysis of preliminary symptomatology. This paper shows the development of a system based on machine learning algorithms for the identification of COVID-19 patients, for which a database of the Kaggle platform (Community of Data Scientists and Machine Learning professionals) was used with a registry of 5,434 people, of which 1,051 were negative cases and 4,383 positive cases for the disease. The work determined the most relevant symptoms, and defined a machine learning algorithm for data classification and analysis, which yielded as performance metrics an F1 score of 0.98 and an aucROC of 0.99. A graphical environment was developed to integrate the model, using Jupyter, QtDesinger and PyQt5, to make it easier for the user to handle the Artificial Neural Network (ANN) model obtained.

Keywords - COVID symptoms, prevention, physiological data, ANN, performance indicators, binary cross entropy.

[1] Ingeniero de Sistemas, UNISANGIL.
[2] Doctor en Tecnología de Avanzada; Docente – Investigador UNISANGIL.

I. INTRODUCCIÓN

Durante los años 2019 y 2020 se presentaron un sin número de eventos de gran importancia para la vida actual, pero sin duda uno de los más grandes hitos fue la llegada de un nuevo virus el cual amenazó la población humana debido a su rápida propagación y contagio. El nuevo virus muy parecido a los causantes de las epidemias SARS en el año 2003 y el MERS en 2012, proveniente de un virus endémico del murciélago, el cual evolucionó hasta llegar a afectar al ser humano, fue nombrado SARS-CoV-2 [1].

Este patógeno se transmite de dos maneras: partículas que quedan suspendidas en el aire llamadas aerosoles y otras partículas que han sido expulsadas por un individuo enfermo las cuales se quedan en las superficies y se transmite por contacto [2].

En Colombia el Ministerio de Salud mediante la circular externa 005 de 2020, estableció los lineamientos a seguir en las diversas Instituciones Prestadoras de Servicios de Salud (IPS) del país. Igualmente, mediante la circular 0017 del mismo año, se definieron las estrategias a seguir por parte de los empleadores y contratantes, a fin de dar respuesta a los posibles contagios presentados en su momento. Por otra parte, se establecieron las pruebas pertinentes (rRT-PCR) que corresponden a las recomendadas por la OMS, para descartar otro tipo de enfermedades respiratorias, como bacterias comunes, las cuales se enviaban al Laboratorio Nacional de Referencia para realizar el análisis [3]. Adicionalmente, se aplicaron las pruebas de detección de antígenos la cual consiste en detectar proteínas del virus a través de una muestra de fluido nasal o bucofaríngeo procesada en un tiempo de 15 minutos, y finalmente la prueba de detección de anticuerpos, la cual se realiza a través de una muestra de sangre para detectar la respuesta inmune contra el virus, esta es procesada en un tiempo de 10 minutos. Sin embargo, existe una necesidad en la sociedad colombiana de poseer más herramientas a disponibilidad que permitan realizar análisis en menor tiempo. Se conocen aplicaciones para el seguimiento de este tipo de infecciones como el Corona Map de la universidad Johns Hopkins, The Coronavirus App de Taiwan e incluso plataformas públicas como *self-quarantine safety* protección del gobierno de Korea. Así mismo, Google ha facilitado el acceso a información generada por la misma movilidad comunitaria, generando así grandes conjuntos de información útiles para el trabajo contra la pandemia. Otros países acompañados por empresas desarrolladoras de aplicativos están trabajando en diferentes tipos de programas los cuales pueden servir como una alternativa para poder controlar la pandemia. Algunos de estos, como se menciona en [4], trabajan con tecnologías como geolocalización, ya sea de un dispositivo móvil o por medio de las redes sociales, y también con pasaportes digitales de inmunidad, y aplicativos que almacenan información para hacer un seguimiento de los contactos. En el trabajo reportado por [5] predicen una infección COVID-19 mediante 8 preguntas básicas de las cuales 5 se refieren a la sintomatología (fiebre, tos, dolor de garganta, dificultad respiratoria y dolor de cabeza) obteniendo una precisión del 90%.

Por otra parte, se han reportado trabajos como el de Miramontes et. al, quienes desarrollaron la plataforma tecnológica denominada *PlaIMoS* para monitorear frecuencia cardíaca y parámetros asociados con enfermedades respiratorias crónicas, de pacientes fuera del entorno clínico de manera eficiente y rentable [6]. También, en [7] los autores proponen una arquitectura que combina computación *Edge* y *Fog*, tecnología LPWAN, IoT y algoritmos de aprendizaje profundo para realizar tareas de monitoreo de salud, en el caso de estudio, utilizando sensores inerciales como entrada tienen una precisión promedio de más del 90% y una recuperación promedio de más del 95% en la detección de enfermedades especificadas.

En este sentido, el proyecto presentado se fundamenta en el desarrollo de herramientas tecnológicas integrales para el fortalecimiento de la capacidad pública de prevención y control de enfermedades, en coherencia con los resultados de aprendizaje del programa de Ingeniería de Sistemas de UNISANGIL.

En este sentido, se muestra el desarrollo de una interfaz gráfica basada en algoritmos de aprendizaje automático para la identificación de pacientes COVID-19; en el desarrollo se integró sintomatología y condiciones clínicas de 5.434 personas, de las cuales 1.051 eran casos negativos y 4.383 casos positivos para la enfermedad, obtenida mediante la plataforma *Kaggle*, para la obtención del modelo. Para el manejo por parte del usuario del modelo de clasificación obtenido, se implementó un entorno gráfico, utilizando *Jupyter*, *QtDesinger* y *PyQt5*, dando facilidad en el diseño y la integración del sistema.

II. MATERIALES Y MÉTODOS

El sistema se desarrolla en lenguaje Python teniendo en cuenta la distribución que se observa en la fig. 1.

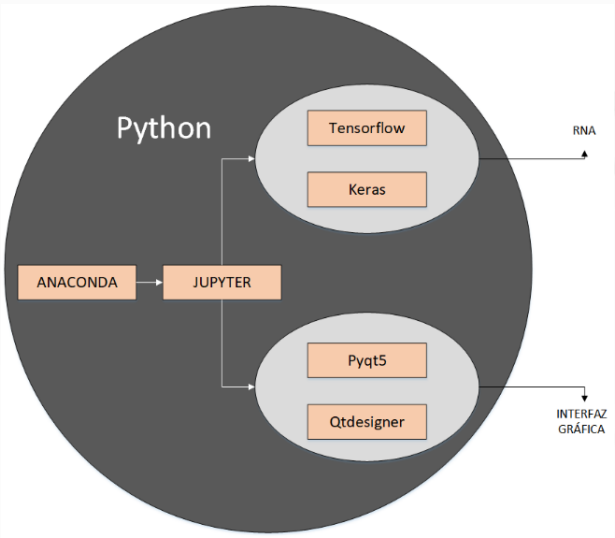

Fig. 1 Diagrama de relaciones del software usado.

Considerando que el trabajo tiene como componentes entrenar una RNA para la clasificación de sintomatología COVID-19 y desarrollar una interfaz gráfica para su manejo, se utiliza la distribución Anaconda y la biblioteca de redes neuronales de código abierto denominada Keras que a su vez se apoya en *TensorFlow*, para obtener una arquitectura lista para ser configurada para la aplicación que se presenta. Por otra parte, se utiliza *QtDesigner* como soporte a PyQt5, que genera código extensión .ui, que al ser convertido a .py posibilita el manejo del código en *Python* en el entorno *Jupyter*. *QtDesigner*, es una herramienta de PyQt5 para crear de manera rápida interfaces gráficas de usuario con widgets del marco Qt GUI. Brinda una interfaz simple de arrastrar y soltar para diseñar componentes como botones, campos de texto, cuadros combinados y más [8].

El lenguaje empleado posee gran capacidad y cantidad de soporte que facilita el desarrollo de futuros proyectos en el orden de análisis y clasificación de información.

Como entorno de programación se utiliza *Jupyter Notebook* el cual es un software que crea un cuaderno *Jupyter* que admite la combinación de código ejecutable, ecuaciones, visualizaciones y texto narrativo [9]. Con Anaconda la cual es una distribución de los lenguajes de programación *Python* y R para computación científica tiene grandes ventajas para simplificar la gestión e implementación de paquetes [10].

A. Modelo de red neuronal artificial utilizado

El modelo de Red Neuronal utiliza *Keras-Tensor*, vinculando características binarias: asma, pulmón crónico, enfermedad cardíaca, diabetes e hipotensión; y seis síntomas clínicos de la enfermedad: Problema respiratorio, fiebre, tos seca, dolor de garganta, secreción nasal y dolor de cabeza de 5.434 personas, donde 1.051 individuos tenían diagnóstico negativo y 4.383 individuos diagnóstico positivo para la enfermedad. Las características del modelo se muestran en la siguiente tabla:

TABLA 1. CARACTERÍSTICAS DE LA RED NEURONAL

Parámetro	Valor
Optimizador	Algoritmo de optimización Adam
Número de nodos en la capa de entrada	11
Número de capas ocultas	2
Número de neuronas por capa oculta	10
Número de nodos en la capa de salida	1
Función de activación en cada capa	Sigmoidea
Función de pérdida	Entropía cruzada binaria
Número de épocas	300
Tamaño de lotes	10

Se utiliza el algoritmo Adam como optimizador el cual se basa en el gradiente de descenso estocástico y ha mostrado gran aplicación en este tipo de modelos en la generación de pesos para las sinapsis de la red. También se elige como función de pérdida al algoritmo de Entropía Cruzada Binaria con la finalidad de conocer la diferencia entre las predicciones realizadas por la RNA y los valores reales. Para validar el desempeño del modelo de red neuronal se realizó una validación cruzada de 5 pliegues, con el índice de desempeño puntaje F1, el cual es una métrica utilizada en problemas con clases desbalanceadas que combina las medidas de precisión y exhaustividad mediante una media armónica. Y el área bajo la curva (aucROC), también es considerado como métrica de desempeño ya que se basa en el cálculo del área de la gráfica de sensibilidad frente a especificidad del modelo clasificador según umbrales de discriminación.

B. Análisis de requerimientos de la interfaz gráfica

Se definen en detalle los requerimientos de la interfaz gráfica de manera que sirvan como especificación del sistema [11].

Requerimientos funcionales

Func-1: El sistema debe permitir el ingreso de un usuario preestablecido, el cual es el que maneja toda la interface de usuario.

Func-2: El sistema debe permitir el ingreso de un usuario preestablecido, el cual va a organizar los posibles síntomas de usuarios.

Func-3: El sistema analizará la información obtenida de los usuarios, contra los parámetros de diagnóstico del algoritmo.

Func-4: El sistema debe suministrar un porcentaje de clasificación positivo o negativo para COVID-19 con base a los síntomas ingresados.

Func-5: El sistema debe crear un reporte con la información que el paciente ingresó, así como el resultado del análisis del sistema.

Requerimientos no funcionales

NFunc-1: El sistema debe estar en capacidad de operar adecuadamente con hasta 14000 datos al mismo tiempo.

NFunc-2: El sistema no revelará a sus operadores otros datos personales de los clientes como nombres y números de referencia.

Requerimientos de usuario

USR-1: El sistema debe contar con una interface amena para el usuario.

USR -2: El sistema debe correr en ambiente Windows 7 o superior.

USR -3: El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario final.

USR -4: El sistema y sus procedimientos de mantenimiento de datos deben cumplir con las leyes y reglamentos de protección de datos médicos.

USR -5: El sistema se diseñará de tal forma que permita una navegación intuitiva.

C. Etapa de diseño de la interfaz gráfica

Considerando los requerimientos establecidos y dando cumplimiento a ellos, se muestran (Fig. 2 y 3) los diagramas UML que ilustran la lógica funcional del sistema.

Fig. 2 Diagrama de casos de uso.

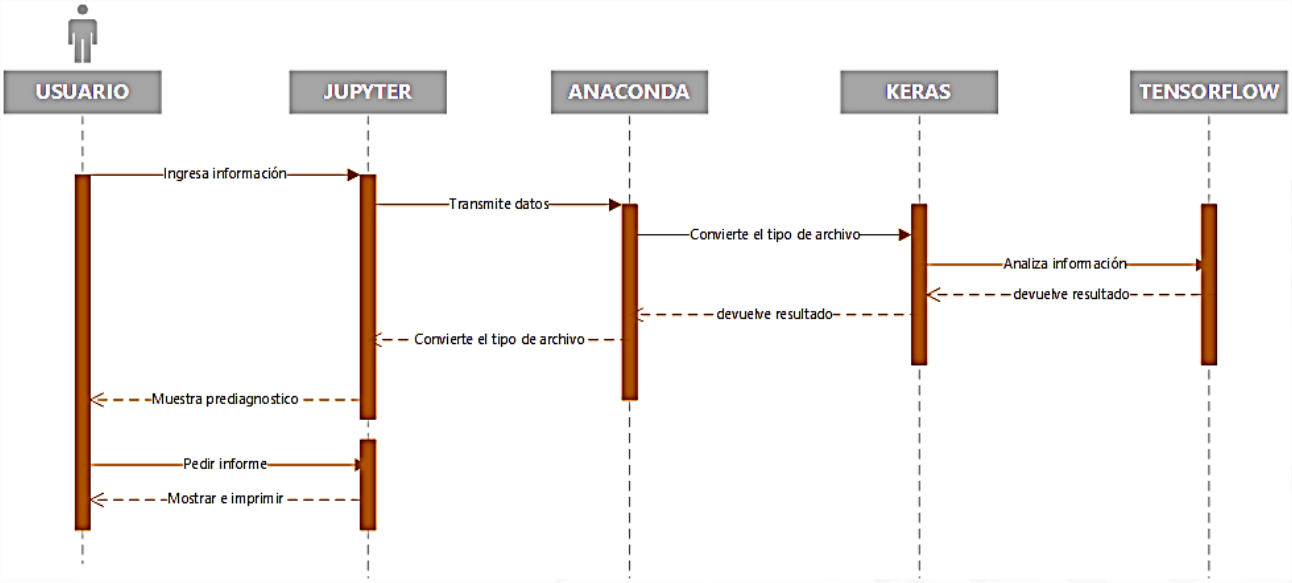

Fig. 3 Diagrama de secuencias.

De esta forma se obtiene una estructura clara para desarrollar el entorno gráfico que permita utilizar el modelo de RNA obtenido para la clasificación de la sintomatología.

III. RESULTADOS Y DISCUSIÓN

A. Desempeño de modelo de red neuronal

El modelo se obtuvo con datos de 5.434 individuos, donde se utilizó el 80% para el entrenamiento y el 20% para prueba. Para garantizar que los resultados son independientes de la partición entre datos de entrenamiento y de prueba, se validó con una validación cruzada de 5 veces, es decir se calcularon 5 modelos con datos aleatorios sacados del mismo conjunto de datos, pero respetando las cantidades en las particiones.

Con Entropía cruzada binara como función de pérdida se obtienen resultados similares en clasificación, de esta forma el índice de desempeño F1 alcanzó un valor de 0.98 con una desviación estándar +/-0.08%, donde valores del índice cercanos a 1 son el mejor escenario, que demuestra un desempeño de clasificación óptimo.

Al considerar el área bajo la curva de ROC como métrica donde se utilizan diferentes umbrales para obtener las tasas de falsos positivos, falsos negativos, el resultado obtenido fue de 0.99, con una variación de ±0.00%. Como se muestra en la Fig. 4 En esta métrica, cuanto más cercano a uno (1) este el valor, mejor desempeño del modelo para diferenciar entre clases positivas y negativas.

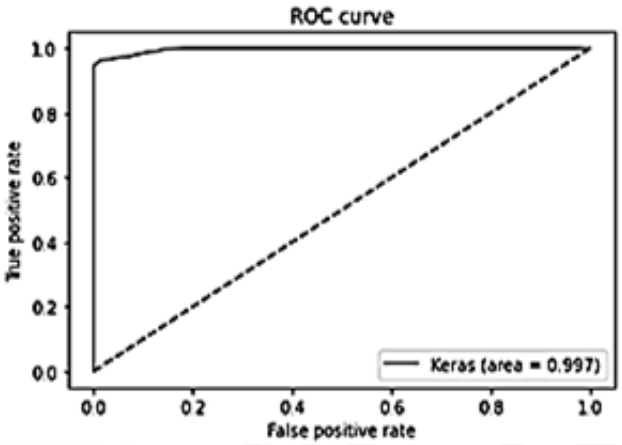

Fig. 4 Curvas ROC que muestran el rendimiento del modelo utilizando la función de pérdida de entropía cruzada binaria.

B. Interfaz gráfica desarrollada

El desarrollo del aplicativo consta de dos elementos, por un lado, el modelo de red neuronal, utilizando *Jupyter* y *Tensor Flow*. Y el entorno o interfaz gráfica, la cual se elaboró utilizando *Jupyter*, *QtDesinger* y *PyQt5*.

Como se observa en la figura 5, la interfaz gráfica posee desplegables para el ingreso de la información relacionada como: Asma, EPOC, Enfermedad Cardiovascular, diabetes e hipertensión, en las cuales el usuario informa al sistema Si tienen o No diagnóstico confirmado de estas. Posteriormente se presenta al usuario una sección en la cual se le solicita el diligenciamiento de la sintomatología relacionada con el diagnóstico de COVID-19, que presenta las cuales son: problemas respiratorios, fiebre, tos seca, dolor de garganta,

dolor de cabeza y congestión nasal, en donde el usuario manifiesta SI presenta estos síntomas o NO.

Posteriormente, el registro de esta información el usuario debe autorizar el uso de esta información a fin de que se habilite la opción de ingreso de datos y seguidamente se habilite la sección de procesamiento de datos que contiene el botón del comando que indica a sistema que proceda a analizar la información, cuyo resultado se presenta en el recuadro llamado probabilidad.

Al integrar el modelo de clasificación en la interfaz gráfica se obtienen dos posibles resultados, un diagnóstico positivo, con su respectiva probabilidad de acierto (Fig. 6), o un diagnóstico negativo y su probabilidad de acierto (Fig. 7).

Igualmente, el usuario tiene la posibilidad de obtener un informe escrito de sus resultados, junto con la información registrada en el sistema, el cual puede usar como soporte para solicitar confirmación del mismo a su médico de confianza (Fig. 8).

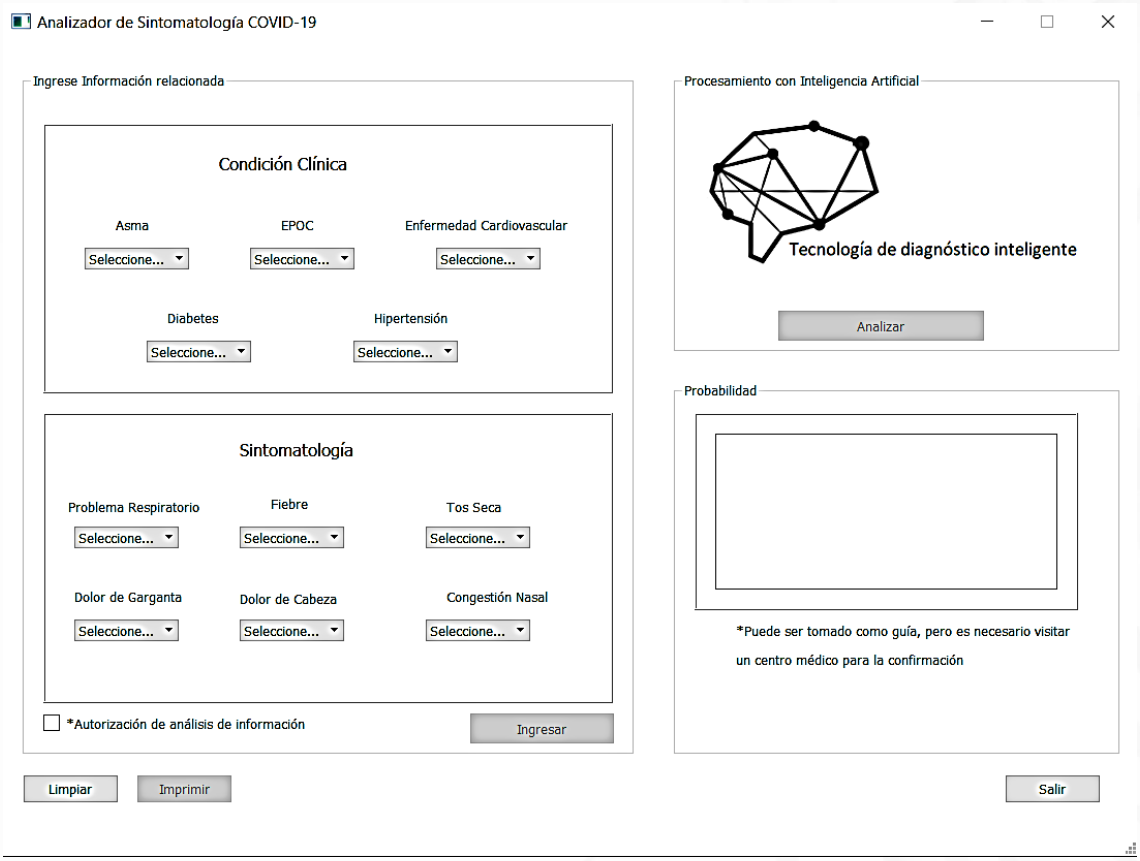

Fig.5 Interfaz gráfica de entrada – Captura de datos.

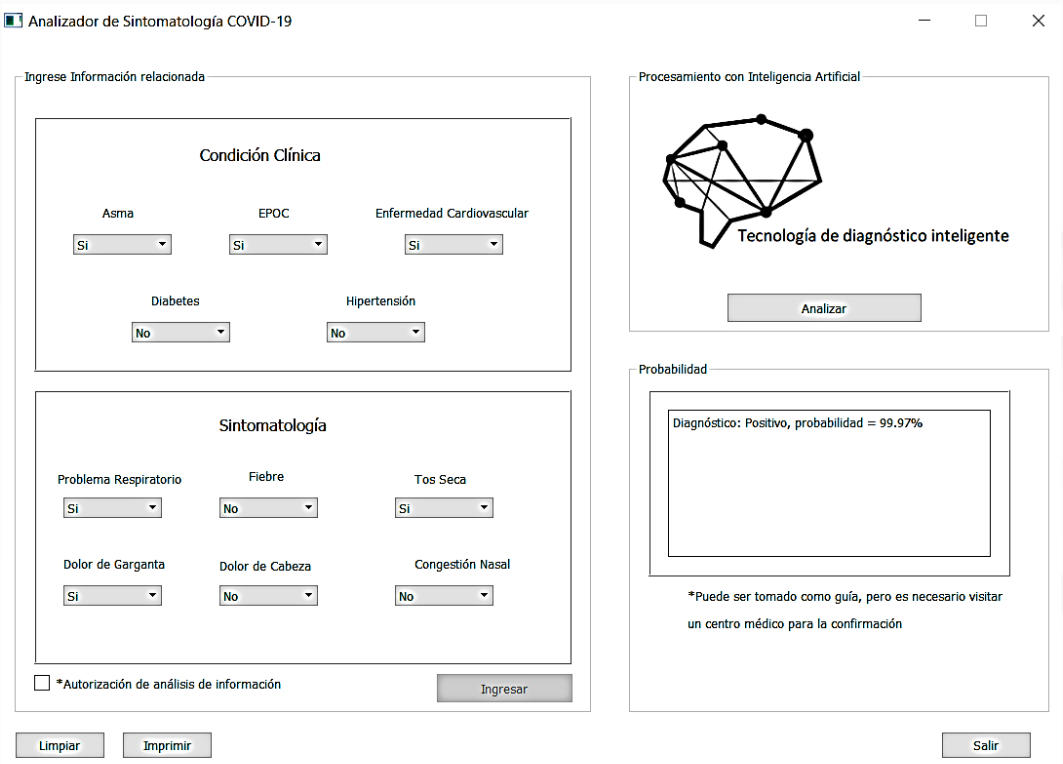

Fig. 6 Interfaz Gráfica con paciente positivo para COVID-19.

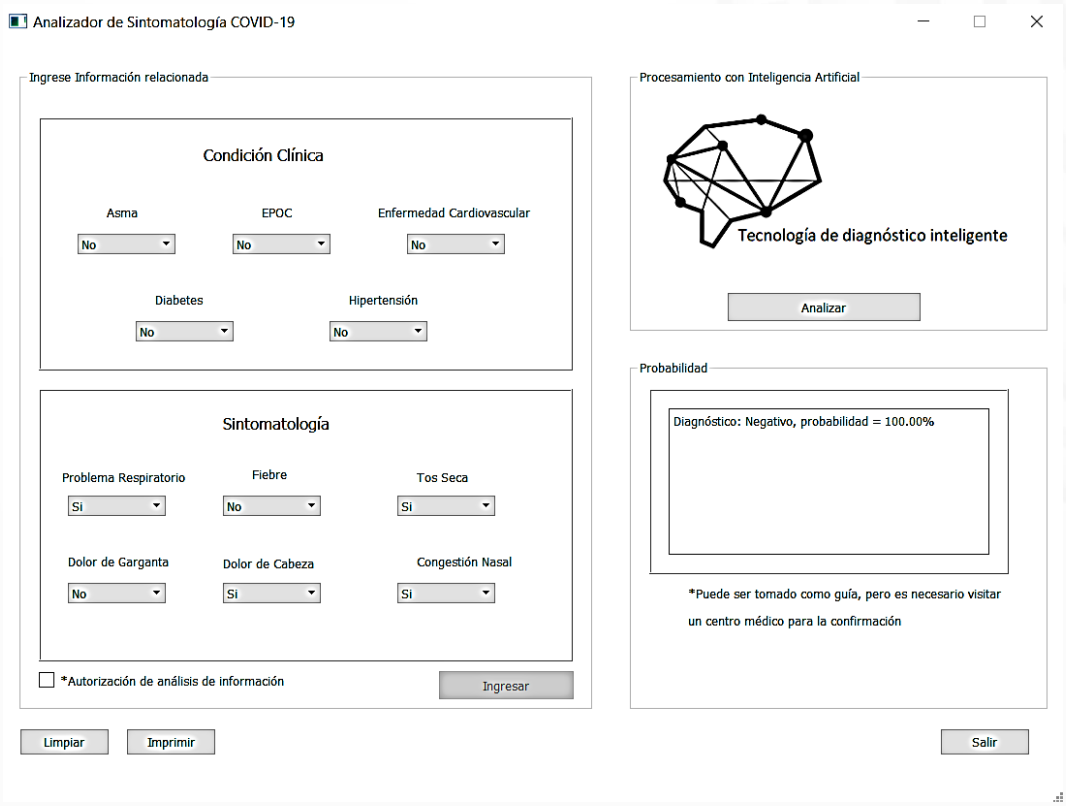

Fig. 7 Interfaz Gráfica con paciente negativo para COVID-19.

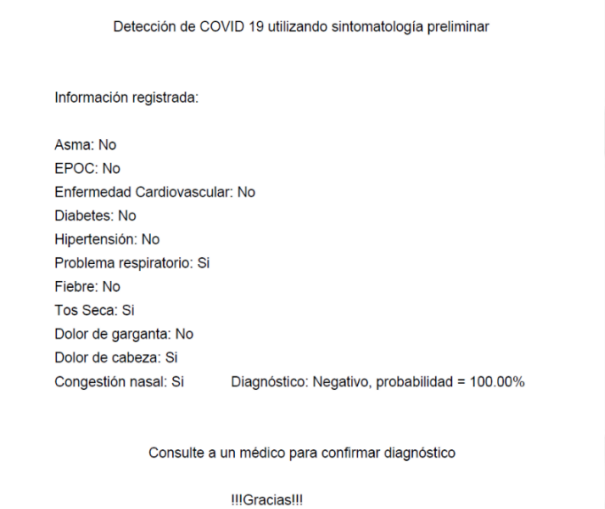

Fig. 8 Impresión de resultados para usuario.

IV. CONCLUSIONES

Luego del proceso de revisión y validación de la información contenida en las diferentes bases de datos públicas disponibles por la OMS, se logró establecer que el conjunto de posibles sintomatologías a utilizar se relaciona con las 11 variables incorporadas al modelo, las cuales se observaron como elementos comunes en la mayoría de ellas.

Como resultado del proceso de desarrollo, se logró un modelo de diagnóstico que luego de realizar las respectivas pruebas de validación, arrojó una puntuación F1 de 0,98 y de 0,99 *aucROC*, las cuales, permiten afirmar que el modelo cuenta con un porcentaje adecuados en el proceso de detección de posibles casos de COVID-19

El sistema sirve como base para el desarrollo de aplicaciones que puedan beneficiar la respuesta de los sistemas sanitarios ante esta enfermedad y otros virus respiratorios, aunque es necesario disponer de datos más robustos que complementen el estudio y eviten posibles sesgos.

AGRADECIMIENTOS

Los autores agradecen a la Fundación Universitaria de San Gil UNISANGIL, por el apoyo en la realización de este trabajo, mediante el proyecto de convocatoria interna CI-03-2019.

REFERENCIAS

[1] E. M. Amenta, A. Spallone, M. C. Rodriguez-Barradas, H. M. El Sahly, R. L. Atmar y P. A. Kulkarni, "Postacute COVID-19: An Overview and Approach to Classification," Open Forum Infectious Diseases, vol. 7, no. 12, pp. 1–7, 2020. [Online]. Available: https://doi.org/10.1093/ofid/ofaa509.

[2] Maldonado, "El contagio de COVID-19 en espacios abiertos," pp. 1–2, 2020. [Online]. Available: https://uniandes.edu.co/es/noticias/salud-y-medicina/riesgo-contagio-coronavirus-espacios-abiertos.

[3] Instituto Nacional de Salud, "Coronavirus Colombia," [Online]. Available: https://www.ins.gov.co/Noticias/Paginas/Coronavirus.aspx.

[4] Organización Internacional del Trabajo, "PREVENCIÓN Y MITIGACIÓN DEL COVID-19 EN EL TRABAJO," 2020.

[5] Y. Zoabi, "COVID-19 diagnosis prediction by symptoms of tested individuals: a machine learning approach," May 2020. [Online]. Available: https://doi.org/10.1101/2020.05.07.20093948.

[6] R. Miramontes et al., "PlaIMoS: A remote mobile healthcare platform to monitor cardiovascular and respiratory variables," Sensors (Switzerland), vol. 17, no. 1, pp. 1–24, 2017. [Online]. Available: https://doi.org/10.3390/s17010176.

[7] J. P. Queralta, T. N. Gia, H. Tenhunen y T. Westerlund, "Edge-AI in LoRa-based health monitoring: Fall detection system with fog computing and LSTM recurrent neural networks," in 2019 42nd International Conference on Telecommunications and Signal Processing, TSP 2019, pp. 601–604. [Online]. Available: https://doi.org/10.1109/TSP.2019.8768883.

[8] The QT Company, "Qt Designer Manual," QT Documentation, 2022. [Online]. Available: https://doc.qt.io/qt-6/qtdesigner-manual.html.

[9] L. A. Barba et al., "Teaching and Learning with Jupyter," 2019. [Online]. Available: https://jupyter4edu.github.io/jupyter-edu-book/.

[10] Rondón, "¿Qué es Anaconda? -Escuela Internacional de Posgrados," 2022. [Online]. Available: https://eiposgrados.com/blog-python/que-es-anaconda/.

[11] L. Flesia et al., "Predicting Perceived Stress Related to the Covid-19 Outbreak through Stable Psychological Traits and Machine Learning Models," n.d.