

Prototipo4

Derly Yanneth Rojas Herrera, Juan Felipe Salamanca González

Abstract—

This report describes the development of a microservice API using FastAPI, designed to detect emotions in images using an artificial intelligence model. It also details the creation of a web application that allows users to upload images and visualize the emotion prediction results. The complete integration between the web application and the microservice API was verified through extensive testing to ensure correct and accurate operation. This development combines modern and affordable technologies to facilitate the implementation of deep learning-based solutions in web applications.

Palabras clave: Microservicio, FastAPI, IA, Detección de Emociones, Aplicación Web, Modelo de Deep Learning.

I. INTRODUCTION

El reconocimiento de emociones a partir de imágenes es un campo de creciente importancia gracias a los avances en inteligencia artificial y aprendizaje profundo. Este proyecto tiene como objetivo desarrollar una solución completa que permita a los usuarios cargar imágenes a través de una aplicación web y recibir una predicción de las emociones presentes en ellas. Para ello, se implementa un microservicio API con FastAPI que utiliza un modelo de deep learning entrenado en Google Colab, y una aplicación web que interactúa con este microservicio.

II. RESULTADOS

Parte 1: Creación de la API con FastAPI Configuración del Entorno:

Se comenzó creando un entorno virtual para Python, asegurando un espacio aislado para las dependencias del proyecto. Se instalaron FastAPI y Uvicorn, junto con bibliotecas adicionales necesarias para ejecutar el modelo de inteligencia artificial.

Desarrollo de la API:

El proyecto de FastAPI se inició con la creación de un endpoint POST que permite la recepción de imágenes. El modelo de IA entrenado se cargó y preparó para la inferencia. Cuando se recibe una imagen, se procesa y se utiliza el modelo para predecir las emociones presentes. Los resultados de la predicción se devuelven como respuesta del endpoint.

Parte 2: Desarrollo de la Aplicación Web Estructura de la Aplicación Web:

La aplicación web se estructuró utilizando archivos base de HTML, CSS y JavaScript. Bootstrap se utilizó para diseñar una interfaz de usuario atractiva y responsiva. Se añadió un formulario que permite a los usuarios subir imágenes y un

botón para enviar estas imágenes al microservicio API para su predicción.

Interacción con la API:

jQuery se empleó para manejar la acción de cargar la imagen y realizar la solicitud POST al microservicio API. Los resultados de la predicción se mostraron en la interfaz de usuario de manera clara y accesible.

Parte 3: Integración y Pruebas Integración Completa:

Se verificó que la aplicación web y el microservicio API estuvieran correctamente conectados. Se probó el flujo completo: carga de imagen, predicción y visualización de resultados para asegurar un funcionamiento sin errores.

Pruebas y Depuración:

Se realizaron pruebas con diversas imágenes para asegurar la precisión del modelo en la detección de emociones. Durante las pruebas, se identificaron y corrigieron problemas, asegurando que la aplicación funcionara de manera robusta y precisa.

III. EVIDENCIAS

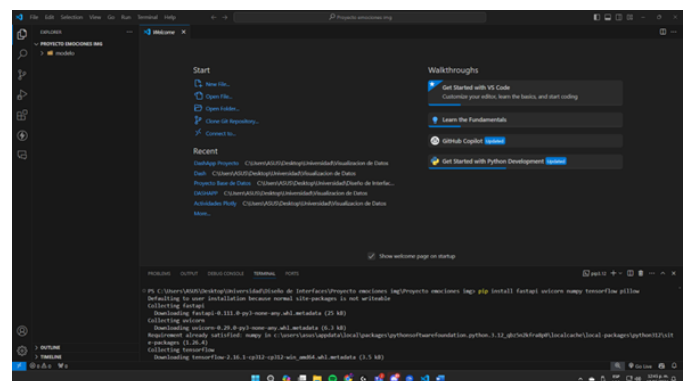


Fig. 1. Instalación de: pip fastapi, uvicorn, numpy, tensorflow y pillow

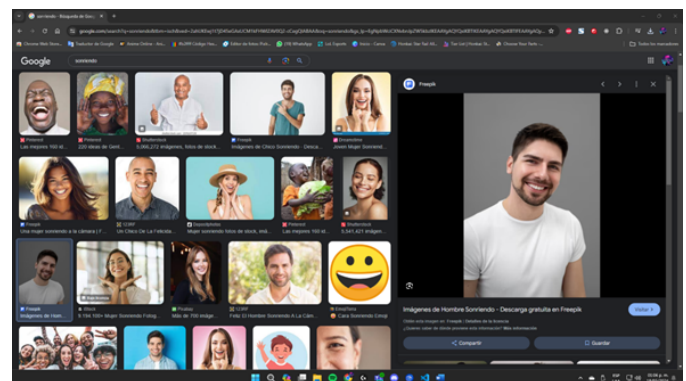


Fig. 2. Búsqueda de imágenes para el modelo

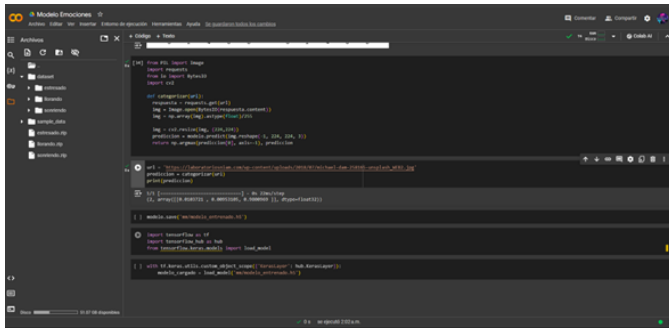


Fig. 3. Probando el entrenamiento del modelo

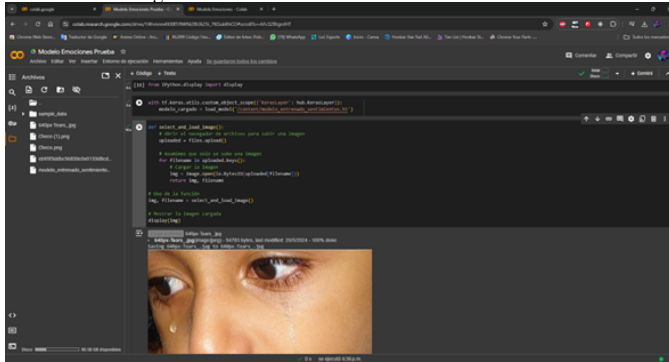


Fig. 4. Probando el modelo y obtención de resultados

IV. CONCLUSIÓN

El proyecto demostró la viabilidad de integrar un modelo de inteligencia artificial para la detección de emociones en imágenes con un microservicio API y una aplicación web. La elección de FastAPI permitió un desarrollo rápido y eficiente del backend, mientras que el uso de Bootstrap y jQuery facilitó la creación de una interfaz de usuario intuitiva y responsiva. Las pruebas confirmaron la precisión del modelo y la robustez de la integración, validando así la efectividad de la solución propuesta.