

Prototipo 4 - Análisis de sentimientos de imágenes

Andrés Felipe Patarroyo Muñoz - 506221032

Santiago Jair Torres Rivera - 506221074

Abstract—Este documento presenta la implementación de un sistema de evaluación de emociones basado en IA, utilizando técnicas de aprendizaje profundo para analizar imágenes y determinar las emociones expresadas en ellas. Dividido en backend (con FastAPI) y frontend, permite a los usuarios cargar imágenes y obtener predicciones de emociones en tiempo real. Esta aplicación destaca el potencial de la IA para mejorar la comprensión emocional en el mundo digital actual.

Keywords: Inteligencia artificial, reconocimiento de emociones, aprendizaje profundo, FastAPI, TensorFlow, backend, frontend, predicción de emociones, interfaz de usuario, análisis de imágenes.

I. INTRODUCCIÓN

En la era digital actual, la integración de la inteligencia artificial (IA) en diversas aplicaciones ha revolucionado la forma en que interactuamos con la tecnología. En este contexto, el desarrollo de sistemas de reconocimiento de emociones basados en IA ha ganado relevancia, permitiendo a las aplicaciones interpretar y responder a las emociones humanas de manera inteligente. Este documento presenta una implementación detallada de un sistema de evaluación de emociones basado en IA, que utiliza técnicas de aprendizaje profundo para analizar imágenes y determinar las emociones expresadas en ellas.

La implementación se divide en tres componentes principales: el entorno virtual y las bibliotecas utilizadas, el backend desarrollado con FastAPI, y el frontend diseñado para proporcionar una interfaz intuitiva a los usuarios. En el entorno virtual, se destacan las diversas bibliotecas utilizadas, incluida TensorFlow para el modelo de IA, y se aborda el manejo de conflictos durante la instalación. El backend, construido con FastAPI, ofrece endpoints para cargar imágenes y obtener predicciones de emociones utilizando el modelo de IA. Por último, el frontend proporciona una experiencia interactiva, permitiendo a los usuarios cargar imágenes, recibir retroalimentación visual instantánea y ver los resultados de las predicciones de emociones en tiempo real.

II. IMPLEMENTACIÓN

Entorno virtual y librerías

Primeramente hablaremos de las librerías que tuvimos que instalar en nuestro entorno virtual, el cual será un entorno aislado para este proyecto, esta es la lista de librerías usadas: numpy, namex, absl-py, annotated-types, anyio, astunparse, cachetools, certifi, charset-normalizer, click, colorama, distlib, dnspython, email-validator, fastapi, fastapi-cli, filelock, flatbuffers, gast, google-auth, google-auth-oauthlib, google-pasta, grpcio, h11, h5py, httpcore, httptools, httpx, idna, Jinja2, keras,

libclang, Markdown, markdown-it-py, MarkupSafe, mdurl, ml-dtypes, oauthlib, opt-einsum, optree, orjson, packaging, Pillow, pip, platformdirs, protobuf, pyasn1, pyasn1-modules, pydantic, pydantic-core, Pygments, python-dotenv, python-multipart, PyYAML, requests, requests-oauthlib, rich, rsa, setuptools, shellingham, six, sniffio, starlette, tensorboard, tensorboard-data-server, tensorflow, tensorflow-estimator, tensorflow-hub, tensorflow-intel, tensorflow-io-gcs-filesystem, termcolor, tf-keras, typer, typing-extensions, ujson, urllib3, uvicorn, virtualenv, watchfiles, websockets, Werkzeug, wheel, wrapt, tensorflow[and-cuda], cabe aclarar que había conflicto con la carga del modelo, e instalando la librería específica de tensorflow[and-cuda] se solucionó, incluso si el modelo no era cargado con una tarjeta gráfica Nvidia que haga uso del CUDA.

Backend

El backend de esta aplicación, desarrollado con FastAPI, ofrece una serie de endpoints destinados a la predicción de imágenes mediante un modelo de aprendizaje automático.

El primer endpoint, accesible mediante la ruta `"/"`, devuelve una plantilla HTML que contiene un formulario de carga de archivos de imágenes. Esta página principal proporciona al usuario la interfaz para seleccionar y enviar imágenes al servidor con el propósito de obtener predicciones.

El segundo endpoint, definido en la ruta `"/prediccion"`, recibe una imagen como entrada a través de una solicitud POST. Al recibir la imagen, el backend procede a procesarla para que coincida con el formato requerido por el modelo de aprendizaje automático. Primero, la imagen se lee y se convierte en formato RGB. Luego, se redimensiona a 224x224 píxeles para ajustarse a las dimensiones esperadas por el modelo. Posteriormente, se convierte en un array de NumPy y se normaliza dividiendo los valores de píxeles por 255.0.

Una vez preparada la imagen, se realiza una predicción utilizando un modelo de aprendizaje automático previamente cargado. Esta predicción se lleva a cabo mediante el modelo TensorFlow, el cual está disponible en el backend y ha sido entrenado previamente para reconocer patrones en imágenes. La clase predicha se determina como el índice del valor máximo en el vector de predicción obtenido del modelo.

Finalmente, la clase predicha se devuelve como respuesta a la solicitud POST, lo que permite al usuario obtener información sobre el contenido de la imagen cargada.

Frontend

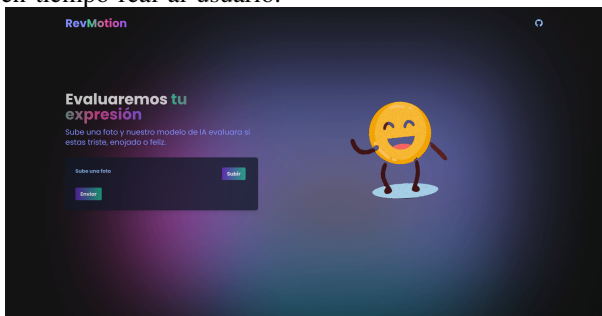
El frontend de esta aplicación ofrece una interfaz intuitiva para que los usuarios puedan interactuar con el sistema de

evaluación de emociones basado en inteligencia artificial (IA). Al cargar la página, los usuarios son recibidos con una cabecera que presenta el nombre del proyecto, "RevMotion", junto con un enlace al repositorio de GitHub para acceder a más detalles sobre el proyecto. Además, la cabecera presenta un fondo degradado que agrega un toque visual atractivo a la página.

El cuerpo principal de la página se divide en dos secciones principales. En primer lugar, el formulario de carga de imágenes, ubicado en el lado izquierdo de la página, invita a los usuarios a "Evaluar tu expresión" y proporciona una breve descripción del propósito del formulario. Aquí, los usuarios pueden seleccionar una imagen desde su dispositivo utilizando un botón de carga de archivos. Una vez seleccionada la imagen, se muestra una vista previa de la misma en la sección derecha de la página, lo que brinda retroalimentación visual inmediata al usuario.

Tras seleccionar una imagen, los usuarios pueden hacer clic en el botón "Enviar" para cargar la imagen al servidor y solicitar una predicción sobre las emociones expresadas en ella. Esta acción activa el segundo componente principal del frontend: la vista previa de la imagen y el resultado de la predicción. Aquí, la sección derecha de la página muestra la vista previa de la imagen seleccionada, permitiendo a los usuarios confirmar visualmente su elección. Justo debajo de la vista previa, un título dinámico espera para mostrar el resultado de la predicción de emociones una vez que se complete el análisis de la imagen.

El archivo JavaScript asociado al frontend desempeña un papel crucial en la experiencia del usuario. Detecta cambios en el formulario de carga de imágenes para mostrar una vista previa instantánea de la imagen seleccionada por el usuario. Posteriormente, envía la imagen seleccionada al servidor para su análisis mediante el uso de solicitudes HTTP. Una vez que el servidor responde con el resultado de la predicción, el JavaScript actualiza dinámicamente el título en la página con la emoción detectada, brindando así una experiencia interactiva y en tiempo real al usuario.



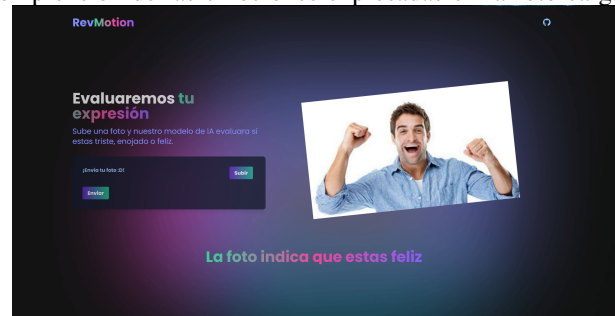
III. RESULTADOS

Al interactuar con la aplicación, el proceso comienza cuando el usuario carga la página web y accede a la funcionalidad de carga de imágenes. Una vez en la página, el usuario selecciona una foto desde su dispositivo y la carga en la aplicación. Esto desencadena un evento en el frontend, donde la imagen seleccionada se muestra como una vista previa en la interfaz

de usuario. Luego, al hacer clic en el botón "Enviar", se activa una solicitud HTTP POST que envía la imagen al servidor para su procesamiento.

En el backend, el servidor recibe la imagen y utiliza un modelo de inteligencia artificial previamente entrenado para analizar las emociones expresadas en la foto. Este modelo, que ha sido entrenado para reconocer patrones en imágenes y asociarlos con emociones específicas, realiza una predicción sobre las emociones presentes en la imagen. La imagen se procesa utilizando técnicas de aprendizaje profundo, donde se extraen características relevantes y se utilizan para hacer la predicción.

Una vez que se completa el análisis, el servidor devuelve el resultado de la predicción al frontend como una respuesta HTTP. El JavaScript en el frontend recibe esta respuesta y actualiza dinámicamente la interfaz de usuario para mostrar el resultado de la predicción. Esto se logra modificando el contenido de la página para incluir el resultado de la predicción, que puede ser una indicación de las emociones detectadas en la imagen, como felicidad, tristeza o estrés. Esta retroalimentación proporcionada al usuario permite una mejor comprensión de las emociones expresadas en la foto cargada.



IV. CONCLUSIÓN

En conclusión, la implementación de este sistema de evaluación de emociones basado en IA ofrece una solución eficaz y fácil de usar para interpretar las emociones expresadas en imágenes. La integración de tecnologías como FastAPI en el backend y JavaScript en el frontend garantiza una experiencia fluida para los usuarios, mientras que el uso de TensorFlow permite realizar predicciones precisas y confiables. Con esta aplicación, los usuarios pueden cargar imágenes y obtener insights sobre las emociones capturadas en ellas, lo que abre nuevas posibilidades en campos como la salud mental, el análisis de redes sociales y la interacción humano-computadora. En última instancia, esta implementación destaca el potencial de la IA para mejorar nuestra comprensión y comunicación emocional en el mundo digital de hoy.