Flagging

Description

A Gradio Interface includes a 'Flag' button that appears underneath the output. By default, clicking on the Flag button sends the input and output data back to the machine where the gradio demo is running, and saves it to a CSV log file. But this default behavior can be changed. To set what happens when the Flag button is clicked, you pass an instance of a subclass of *FlaggingCallback* to the *flagging_callback* parameter in the *Interface* constructor. You can use one of the *FlaggingCallback* subclasses that are listed below, or you can create your own, which lets you do whatever you want with the data that is being flagged.

SimpleCSVLogger

```
gradio.SimpleCSVLogger(···)
```

Description

A simplified implementation of the FlaggingCallback abstract class provided for illustrative purposes. Each flagged sample (both the input and output data) is logged to a CSV file on the machine running the gradio app.

Example Usage

```
import gradio as gr
def image_classifier(inp):
    return {'cat': 0.3, 'dog': 0.7}
demo = gr.Interface(fn=image_classifier, inputs="image", outputs="label",
                    flagging_callback=SimpleCSVLogger())
```

CSVLogger

`dio.CSVLogger(···)`

Description

The default implementation of the FlaggingCallback abstract class. Each flagged sample (both the input and output data) is logged to a CSV file with headers on the machine running the gradio app.

Example Usage

```
import gradio as gr
def image_classifier(inp):
    return {'cat': 0.3, 'dog': 0.7}
demo = gr.Interface(fn=image_classifier, inputs="image", outputs="label",
                    flagging_callback=CSVLogger())
```

Guides

Using Flagging

HuggingFaceDatasetSaver

`gradio.HuggingFaceDatasetSaver(hf_token, dataset_name, ···)`

Description

A callback that saves each flagged sample (both the input and output data) to a HuggingFace dataset.

Example Usage

```
import gradio as gr
hf_writer = gr.HuggingFaceDatasetSaver(HF_API_TOKEN, "image-classification-mistakes")
def image_classifier(inp):
    return {'cat': 0.3, 'dog': 0.7}
demo = gr.Interface(fn=image_classifier, inputs="image", outputs="label",
                    allow_flagging="manual", flagging_callback=hf_writer)
```

Initialization

Parameter	Description

ımeter	Description
`hf_token` *str* *required*	The HuggingFace token to use to create (and write the flagged sample to) the HuggingFace dataset (defaults to the registered one).
`dataset_name` *str* *required*	The repo_id of the dataset to save the data to, e.g. "image-classifier-1" or "username/image-classifier-1".
`private` *bool* **default: False**	Whether the dataset should be private (defaults to False).
`info_filename` *str* **default: "dataset_info.json"**	The name of the file to save the dataset info (defaults to "dataset_infos.json").
`separate_dirs` *bool* **default: False**	If True, each flagged item will be saved in a separate directory. This makes the flagging more robust to concurrent editing, but may be less convenient to use.

Guides

Using Flagging