Hosting a Vector Database with Hugging Face

Here’s a fun app idea: **You Know What They Say**.
The user gives the app a sentence, and the app returns a relevant proverb.

Example:

```
User > "My friend Tommy broke his computer while trying to install a better battery!"
App  > "Well, you know what they say: If it ain't broke, don't fix it!"

User > "I went to the beach before sunrise but all the good spots were taken."
App  > "Well, you know what they say: Early bird gets the worm."
```

We can implement this app by storing hundreds (or even thousands) of English proverbs in a vector database, running a semantic search with the user’s input sentence, and then fetching and returning the most relevant proverb.

Now, what if we wanted to store the data for our app online and do semantic search via HTTP calls instead of locally?

Guess what - Hugging Face allows us to host and query vector datasets online! Let’s see how below.

First, we need to export our vector database to a `.csv` file. We can do this by using `pandas.to_csv()` as follows:

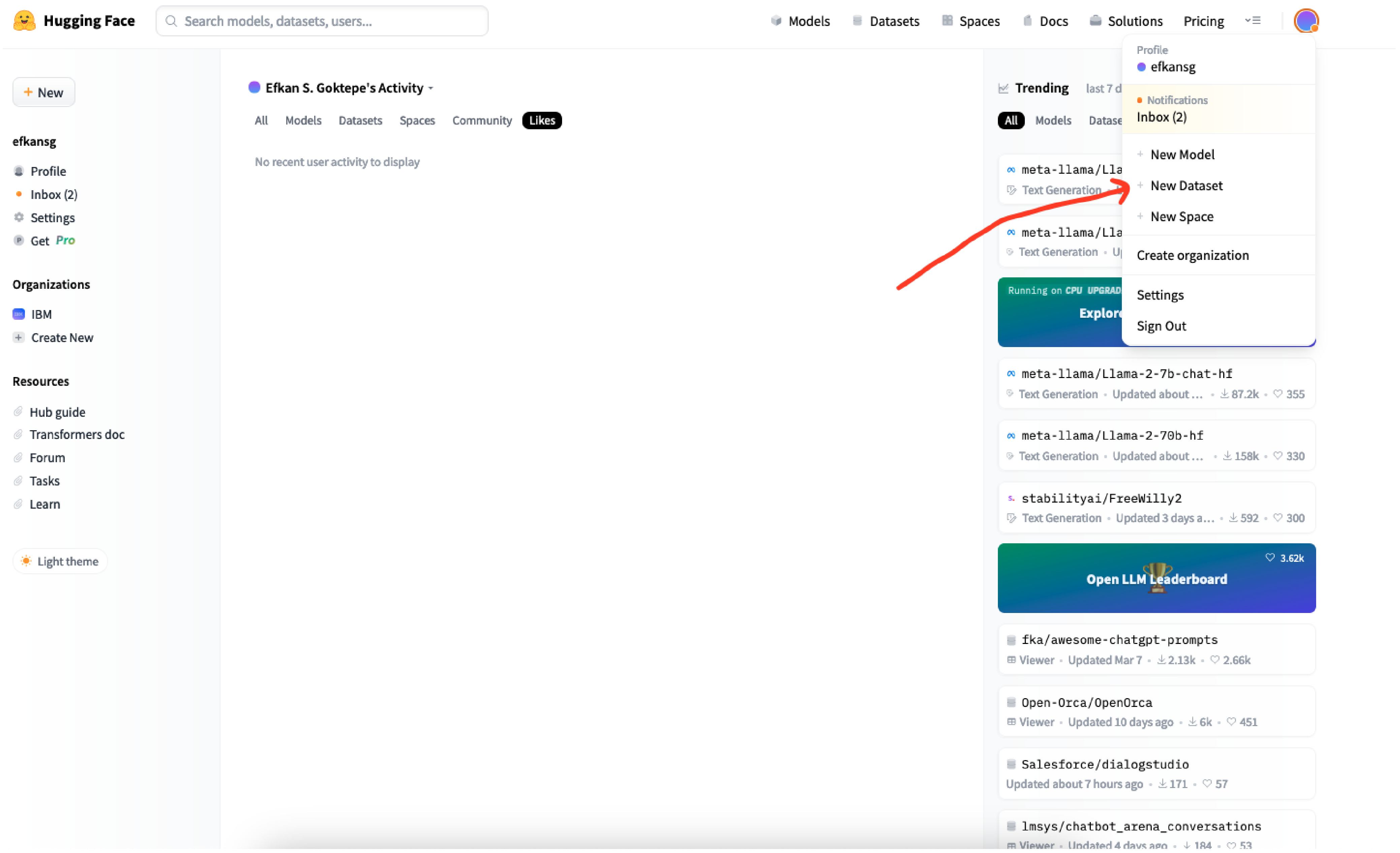
```python
# Convert only the embeddings to a DataFrame, without the texts
df_embeddings = pd.DataFrame(embeddings)

# Export the dataframe as a .csv file
df_embeddings.to_csv("embeddings.csv", index=False)
```

After doing so, you should see `embeddings.csv` in your IDE file explorer.

Next, create a new dataset on Hugging Face by doing the following:

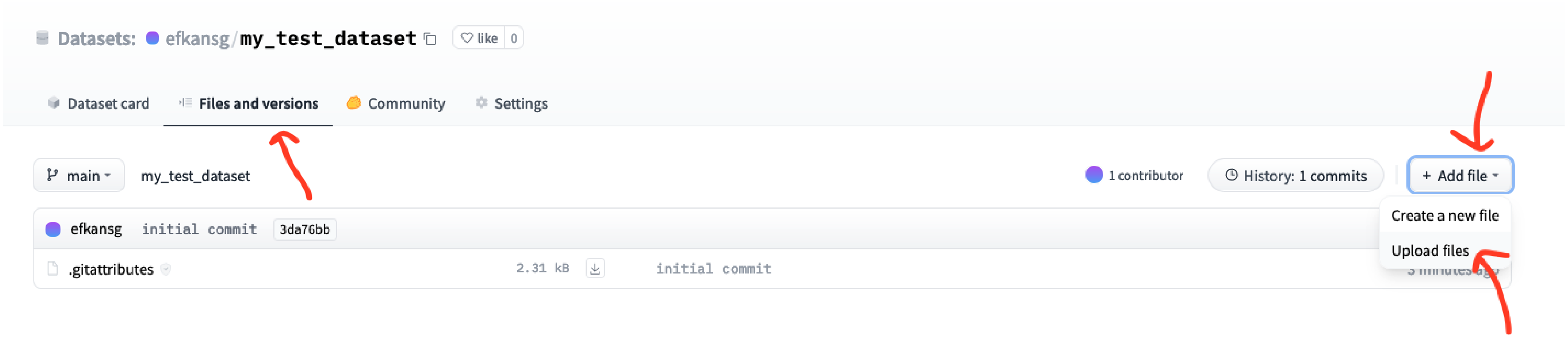
1. Navigate to https://huggingface.co/
2. Click on your user picture in the right-top

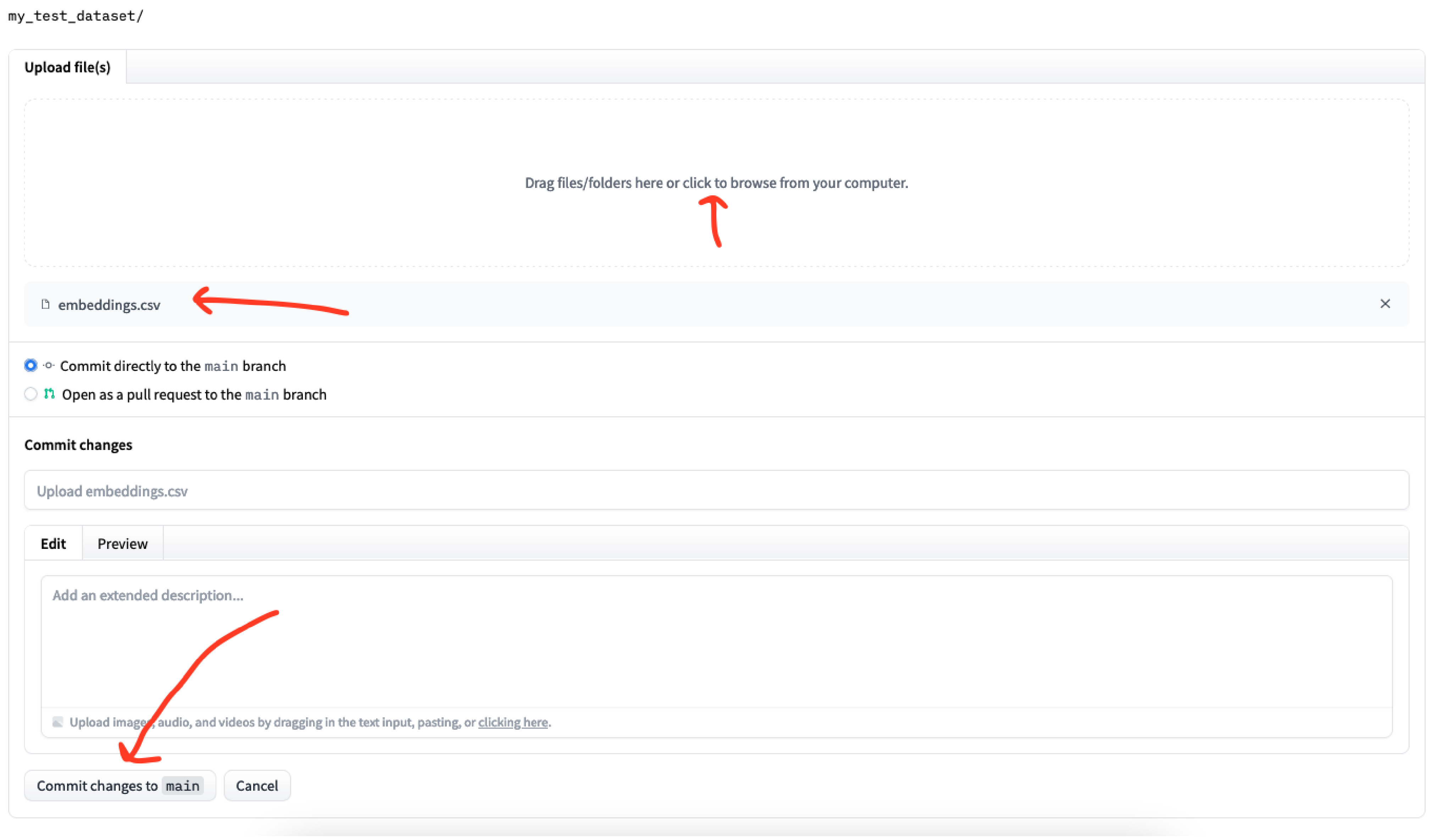
3. Click “New Dataset”
4. Give your dataset a name (e.g. “my_test_dataset”)
5. Make sure the “Public” box is checked
6. Hit “Create dataset”



Then, you should be redirected to your dataset. Add your `.csv` file to the dataset by doing:

1. Navigate to the “Files and Versions” section of your dataset
2. Click “Add File -> Upload files”
3. Drag and drop your files into the box, and hit “Commit changes to `main`“




my_test_dataset/

Upload file(s)

Drag files/folders here or click to browse from your computer.

embeddings.csv

[✓ Commit directly to the main branch]
Open as a pull request to the main branch

Commit changes

Upload embeddings.csv

Edit | Preview

Add an extended description...

Upload images, audio, and videos by dragging in the text input, pasting, or clicking here.

Commit changes to main | Cancel

After uploading our dataset, we can load the embedded dataset from Hugging Face using the `datasets` library and convert it to a PyTorch FloatTensor, which is one way to operate on the data. Make sure to replace `namespace/repo_name` with your user and repo name.

```python
import torch
from datasets import load_dataset
```

```

articles_embeddings = load_dataset('namespace/repo_name')
dataset_embeddings = torch.from_numpy(articles_embeddings["train"].to_pandas().to_numpy()).to(torch.float)
```

Given any article name, suppose we would like to conduct semantic search to generate relevant suggestions. Similar to the previous section, we first get the embeddings for that article.

```
cat_article = ['Cats have very fast reflexes']
response = requests.post(api_url, headers=headers, json={"inputs": cat_article, "options":{"wait_for_model":True}})

query_embeddings = torch.FloatTensor(response.json())
```

Next, we can use the `sentence_transformers` library to query our data. Let's use it to find the most relevant 2 articles.

```
from sentence_transformers.util import semantic_search

# Find top 2 similar vectors using semantic_search
hits = semantic_search(query_embeddings, dataset_embeddings, top_k=2)

# Print result
print(hits)
```

From doing so, you should get the following list:

```
[
    [
        {'corpus_id': 1, 'score': 0.7399995923042297},
        {'corpus_id': 5, 'score': 0.6190879344940186}
    ]
]
```

The `corpus_id` field in each row refers to the index at which the 'similar' vector was found.
We can use this value to index the text from our starting data.

```python
print([examples[hits[0][i]['corpus_id']] for i in range(len(hits[0]))])
```

The output should be:

```
[
    'There are no animals with faster reflexes than cats.',
    'Felines are very quick to react to sudden events.'
]
```