***Now i have such a data my next step is to feed it to an offline ai model to clean it even more, should i make a relational database or only clean the data in the csv file***

**Step-by-Step Guide: Cleaning Reddit Data for Offline AI Model**

**Option 1: Clean Directly in CSV (Simple & Fast)**

**Best for:**

* Quick preprocessing before feeding data to an AI model.
* One-time analysis or small-to-medium datasets.

**Steps:**

1. **Load & Inspect Data**

python

import pandas as pd

df = pd.read\_csv('reddit\_data.csv')

print(df.info()) # Check missing values, dtypes

1. **Drop Irrelevant Columns**  
   Remove columns not needed for your AI model (e.g., gildings, pwls):

python

df = df.drop(columns=['mod\_reason\_title', 'thumbnail\_height', 'wls'])

1. **Handle Missing Data**

python

df['selftext'] = df['selftext'].fillna('') # Replace NaN with empty strings

1. **Filter Rows**  
   Remove deleted/removed posts:

python

df = df[df['removed\_by\_category'].isna()] # Keep only non-removed posts

1. **Standardize Text**  
   Clean title and selftext for NLP:

python

df['cleaned\_text'] = df['title'] + ' ' + df['selftext']

df['cleaned\_text'] = df['cleaned\_text'].str.lower().str.replace(r'[^\w\s]', '')

1. **Save Cleaned CSV**

python

df.to\_csv('cleaned\_reddit\_data.csv', index=False)

**Option 2: Use a Relational Database (Scalable & Structured)**

**Best for:**

* Large datasets (>1GB) or frequent updates.
* Complex queries (e.g., joining with other tables).

**Steps:**

1. **Set Up SQL Database** (e.g., PostgreSQL, SQLite)

bash

# Install PostgreSQL

sudo apt-get install postgresql

1. **Create a Table Schema**

sql

CREATE TABLE reddit\_posts (

id VARCHAR(20) PRIMARY KEY,

title TEXT,

selftext TEXT,

author TEXT,

subreddit TEXT,

created\_utc TIMESTAMP,

score INTEGER,

num\_comments INTEGER,

is\_self BOOLEAN

);

1. **Import CSV to Database**  
   Use psql or Python (e.g., sqlalchemy):

python

from sqlalchemy import create\_engine

engine = create\_engine('postgresql://user:password@localhost:5432/dbname')

df.to\_sql('reddit\_posts', engine, if\_exists='replace')

1. **Clean Data in SQL**

sql

-- Remove removed posts

DELETE FROM reddit\_posts WHERE removed\_by\_category IS NOT NULL;

-- Update empty text

UPDATE reddit\_posts SET selftext = '' WHERE selftext IS NULL;

1. **Export for AI Model**

python

cleaned\_df = pd.read\_sql('SELECT \* FROM reddit\_posts', engine)

cleaned\_df.to\_csv('cleaned\_reddit\_data.csv', index=False)

**Key Considerations**

| **Factor** | **CSV Cleaning** | **Relational Database** |
| --- | --- | --- |
| **Data Size** | <1GB | >1GB |
| **Complexity** | Low | High (joins, indexes) |
| **Scalability** | Limited | High |
| **Maintenance** | Manual | Automated (SQL) |

**Recommendation**

* **Start with CSV Cleaning** if your dataset is small or you need quick results.
* **Use a Database** if you plan to:
  + Continuously update the data.
  + Run complex aggregations (e.g., "Count posts by subreddit").
  + Integrate with other datasets later.

**Example Hybrid Approach**:

1. Clean raw data in CSV/Pandas.
2. Import cleaned data into a database for long-term storage.

***For extra context, now the data is extracted perfectly from reddit, and i extracted the needed columns from the data, now i need to clean the textual unstructured data from the top comment and the title and subtext, so my plan is to use an ready offline LLM to help me clean this data, sadly i dont know what is the needed prompt or what automation tool to use to read the csv file and open the offline ai model and then let the ai model clean it line by line and see if it worth to be included in the final data set that will train THE TARGET LLM. if you have any questions tell me and give me a diagram of the steps that you understood before giving me the reply.***

**Step 1: Identify Key Attributes**

From your dataset, the relevant columns for textual cleaning are:

* title: Post title (short text).
* selftext: Post body (longer text, may be empty for link posts).
* num\_comments: Total comments on the post (use this to filter active discussions).
* top\_comment: *(If available)*. If not, you’ll need to extract it via Reddit API or pushshift.

**Step 2: Diagram of the Workflow**

