**Addressing Issue 2: PyTorch & Streamlit Watcher Error (RuntimeError: no running event loop etc.)**

This error is often related to the environment, how Streamlit's file watcher interacts with complex libraries like PyTorch, or specific versions.

* **No Direct Code Fix in App Logic**: The Python code for your application logic (e.g., in streamlit\_app.py, engine.py) is unlikely to be the direct cause if imports are standard. The problem lies in Streamlit's attempt to monitor PyTorch library files for changes.
* **Common Advice (Environment/Run Configuration)**:
  1. **Update Libraries**: Ensure Streamlit, PyTorch, and Sentence-Transformers are updated to their latest stable versions compatible with your Python version (3.9). An outdated library might have fixed compatibility issues.

Bash

pip install --upgrade streamlit sentence-transformers torch torchvision torchaudio

* 1. **Streamlit Configuration (Advanced)**: You can tell Streamlit not to watch certain folders. This is done via Streamlit's global or project-specific config.toml file. Add the path to your Python environment's torch package to server.folderWatchBlacklist.
     + Find your torch path: python -c "import torch; print(torch.\_\_path\_\_)"
     + Create/edit .streamlit/config.toml in your project root:

Ini, TOML

[server]

folderWatchBlacklist = ["your\_python\_env\_path/site-packages/torch"]

# Add other problematic library paths if needed

* 1. This is often the most effective way to deal with watcher-related errors from specific libraries.
  2. **Run Streamlit with --server.fileWatcherType none**: As a diagnostic step or temporary workaround, you can run Streamlit with file watching disabled, though this means you'll need to manually refresh the browser on code changes.

Bash

streamlit run streamlit\_app.py --server.fileWatcherType none

Since I cannot modify your environment, I will ensure the Python code provided does not do unusual PyTorch imports. The current engine.py lazy-loads SentenceTransformer (which uses PyTorch) inside a cached function, which is good practice.