PTLog

final edited 11/8

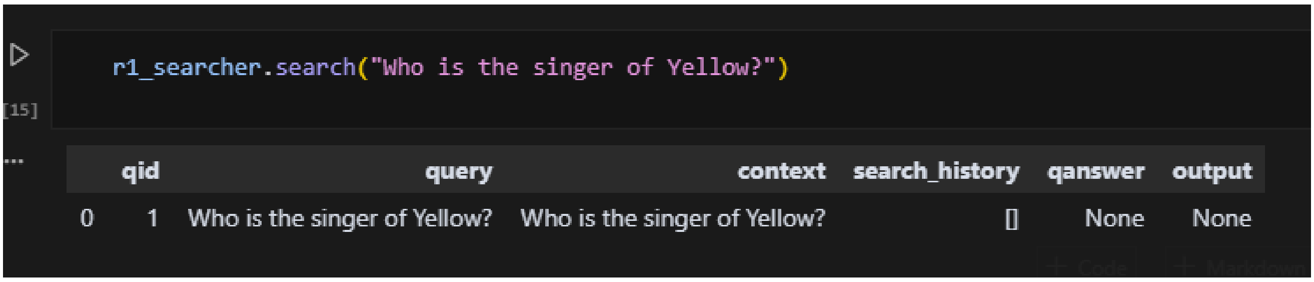
What’s Done

- Standardized all “empty” returns to **`None`** and merged the double-filter step into **one loop**.
- Logged in to Hugging Face via `huggingface-cli login` to enable authenticated model pulls.
- Removed the risk of passing a plain **list of strings** to `.search()`; now build a **DataFrame with `qid`/`query`** and realign results with `groupby("qid")`.
- According to GPT: Because kept failing on using vllm in R1_Searcher, did a backup process to use huggingface backend when failing.

Current Issue

- `R1Searcher.search()` returns empty. Assuming its because the model output isn’t recognized as a **final answer** (no “Final Answer”/tag/`boxed{}` pattern), and it also didn’t emit a **retrieval tag**. The current logic then sets `output` to `None`, so the row looks blank.

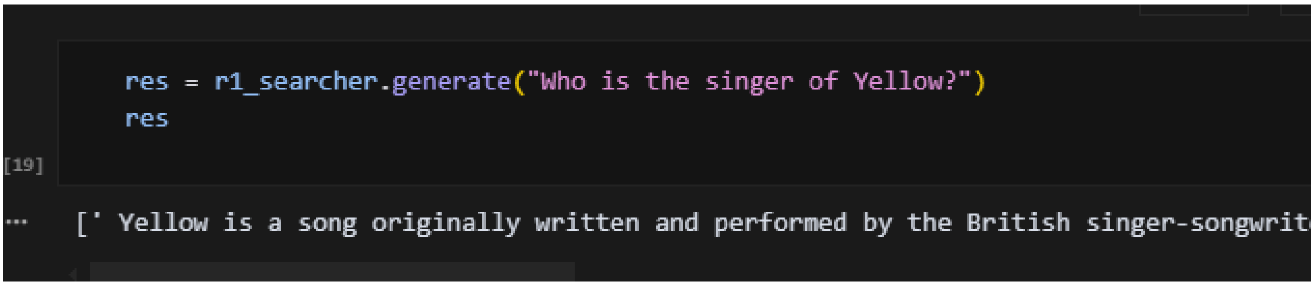
screen shot from frame_test_2.ipynb



```
r1_searcher.search("Who is the singer of Yellow?")
```

	qid	query	context	search_history	qanswer	output
0	1	Who is the singer of Yellow?	Who is the singer of Yellow?	[]	None	None

screen shot from frame_test_2.ipynb



```
res = r1_searcher.generate("Who is the singer of Yellow?")
res
```

```
[' Yellow is a song originally written and performed by the British singer-songwrit
```

Next Steps

- **Broaden** **`format_answers`** patterns and add a **plain-text fallback**: when no tags are present, treat the first sentence as the answer.
- When there's **no retrieval tag but text exists**, **keep the raw generation** and store it in `qanswer` instead of nulling it.
- Tighten the prompt:
  "When confident, reply with: `Final Answer: <answer>` *only."*
  (Improves match rate for the extractor.)