A Reinforcement-based QA Recommender System for Responding to Community-based Suggestions using Enhanced Contextualization

Abhishek Bais, Haley Feng, Princy Joy, Shannon Phu
Advisor: Dr. Ali Arsanjani

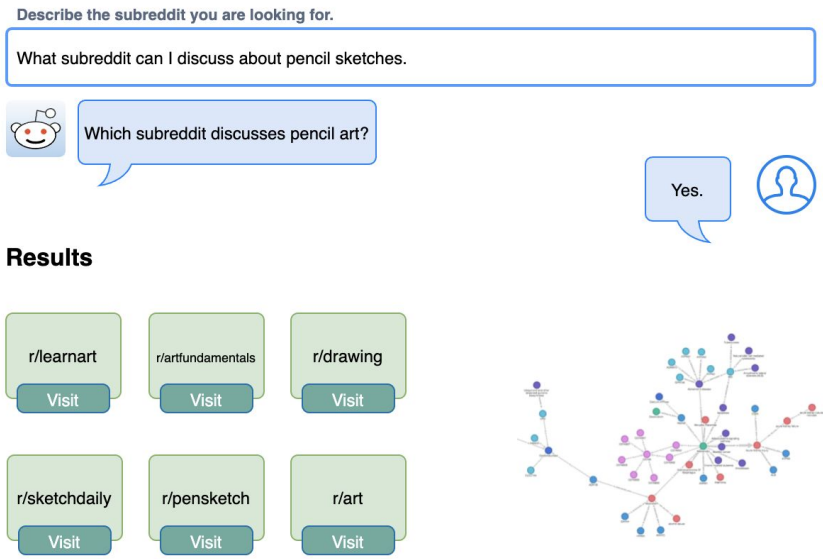
Introduction

- Community-based question-answering social networks provide registered users access to a huge knowledge database of questions and answers accumulated over time
  - ie. Quora, Reddit, StackOverflow, Yahoo Answers, StackExchange
- Users can find an online community to ask, answer, and discuss topics of interest
- In particular, discovering the appropriate group/community is difficult as the social network platform scales to more users
- For example, Reddit has 430M monthly active users and 2.6M+ subreddits

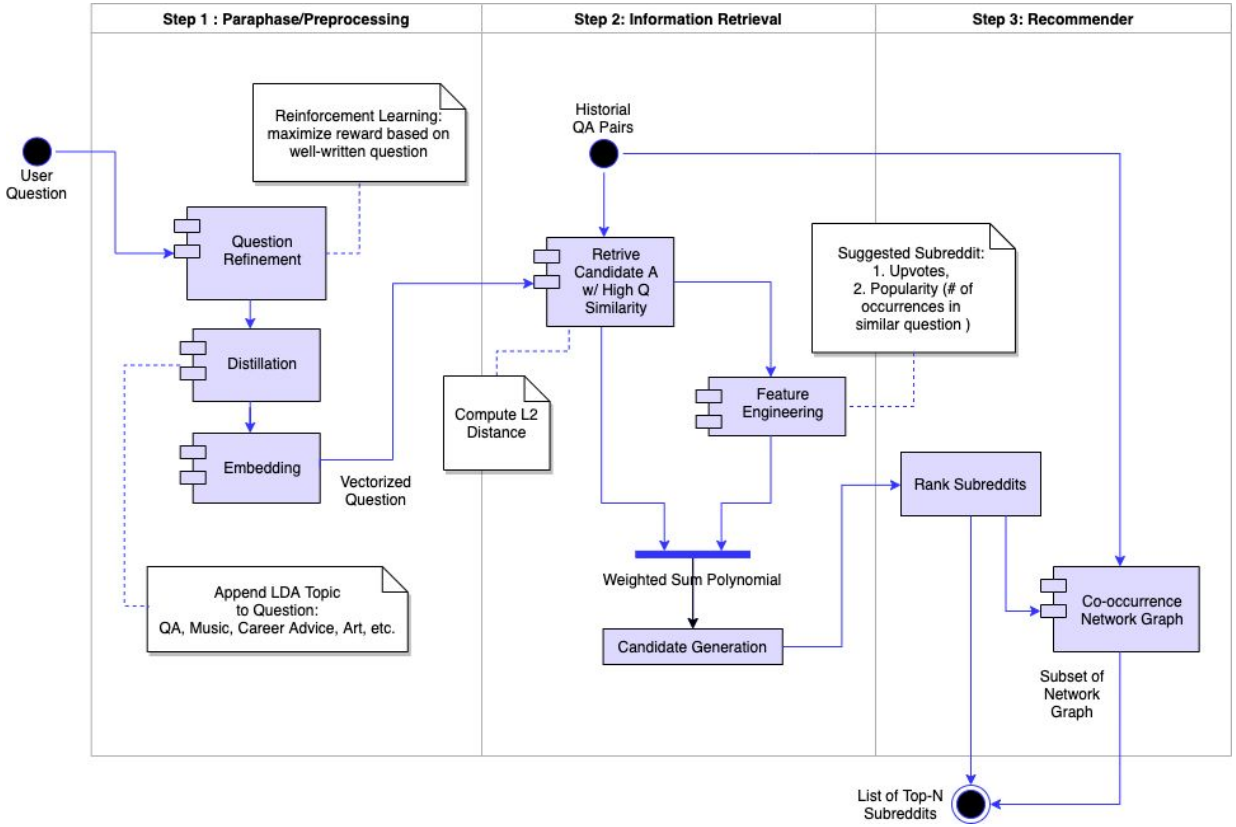
Proposed Solution

- We propose a question-answering machine learning application that:
  - Answers a user's question about what subreddit they are looking for
  - Recommends and ranks subreddits in order of relevance
  - Guides the user to better recommendations through question refinement

<u>Mock:</u>



Architecture

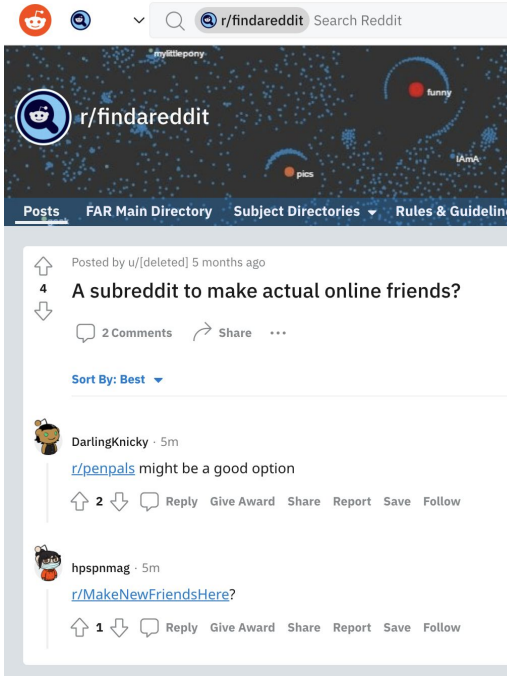


Data Requirements

- In order to train a machine learning model, we need large amounts of training data
- Since we are building a QA application, we needed data with questions and answers
- We need data in the form of
  - Question (what subreddit the user wants to find)
  - Answer (subreddit)
  - Rank score (how good is the answered subreddit)

Data Collection and Processing

- Leveraged Pushshift and PRAW Reddit APIs to scrape r/FindAReddit to get questions and community suggested subreddits
- Scraped 1 year of data 7/1/2020 - 7/1/2021 for 40k total posts
  - Question
  - Answers
  - Upvotes
- Cleaned and extracted only subreddit data

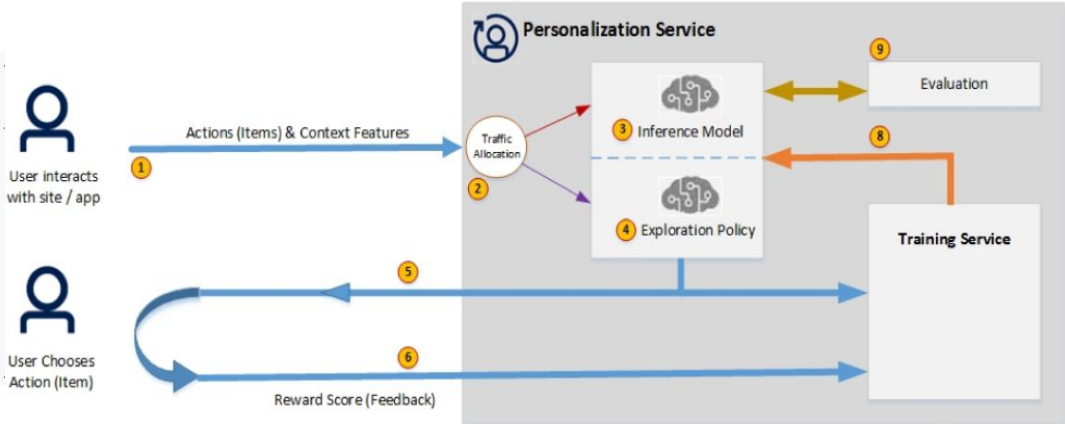


question_id	question	question_vocab	comment_upvotes	suggested_subreddits
o0oakk	A subreddit to make actual online friends?	make actual friend title	2	r/penpals
o0oakk	A subreddit to make actual online friends?	make actual friend title	1	r/makenewfriendshere

Reinforcement-based Question Refining

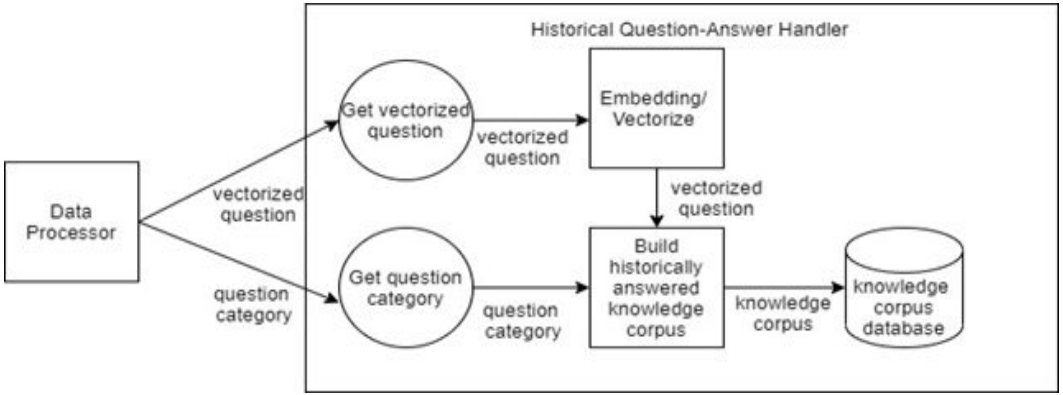
- Purpose: refine the user's question input to the QA system
- Leveraged an open-sourced natural language paraphrase built using the T5 large language model to generate question re-phrasing candidates
- Integrated with Azure Personalizer service which provides online reinforcement learning to learn which question paraphrasing the user prefers

```
-----------------------------------------------------------------
Input_phrase: Can you recommed some upscale restaurants in Newyork?
-----------------------------------------------------------------
list some excellent restaurants to visit in new york city?
what upscale restaurants do you recommend in new york?
i want to try some upscale restaurants in new york?
recommend some upscale restaurants in newyork?
can you recommend some high end restaurants in newyork?
can you recommend some upscale restaurants in new york?
can you recommend some upscale restaurants in newyork?
-----------------------------------------------------------------
```



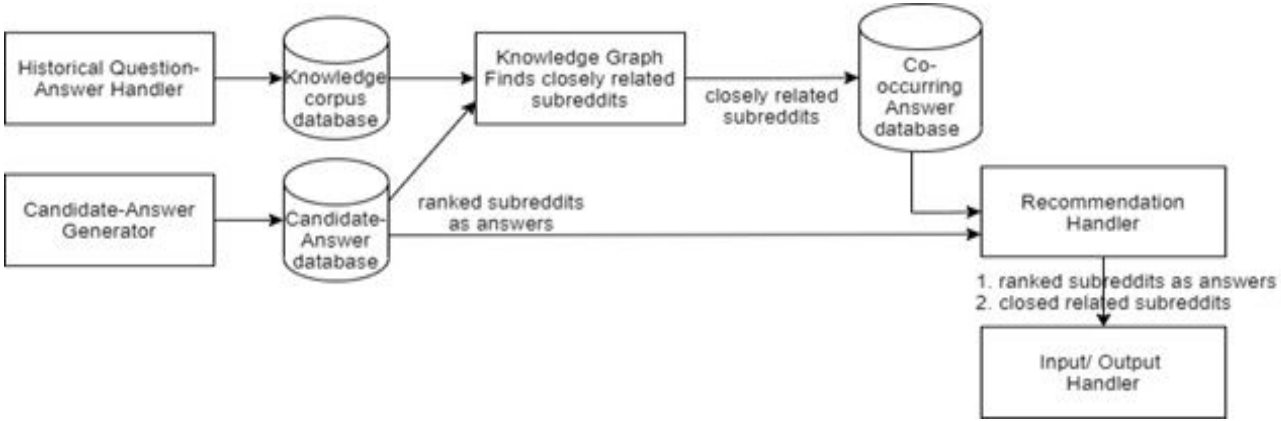
Information Retrieval

- First we obtain embeddings (numerical vector representations) for each questions' text
- We index the embeddings in order to find the best similarity between a new question and the questions in our historical dataset



Subreddit Recommender / Ranker

- We extract candidate subreddits based on similarity from the historical questions index
- We rank the resulting candidate subreddits based on
  - Similarity scores
  - Popularity via upvotes
  - Number of subreddit occurrences among similar questions



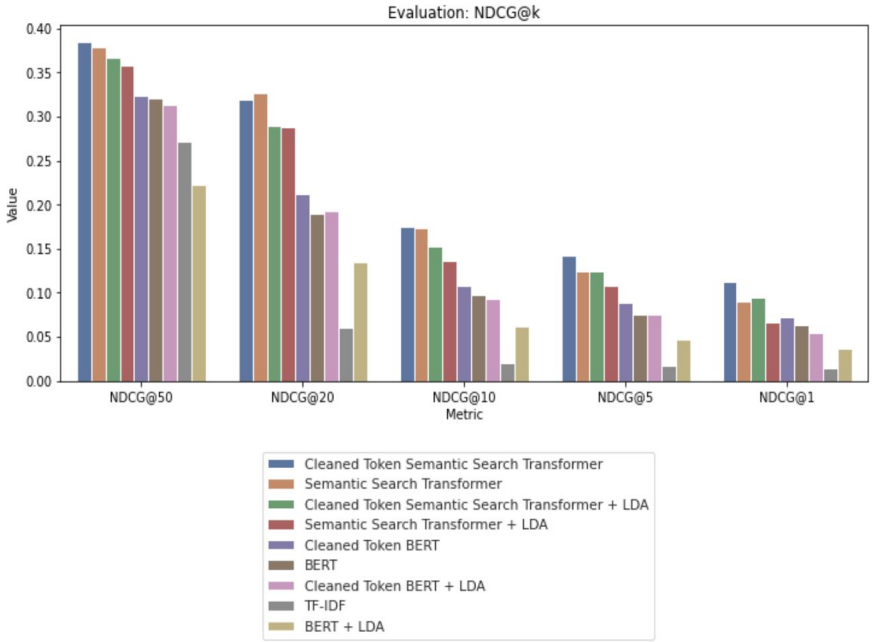
Experiments

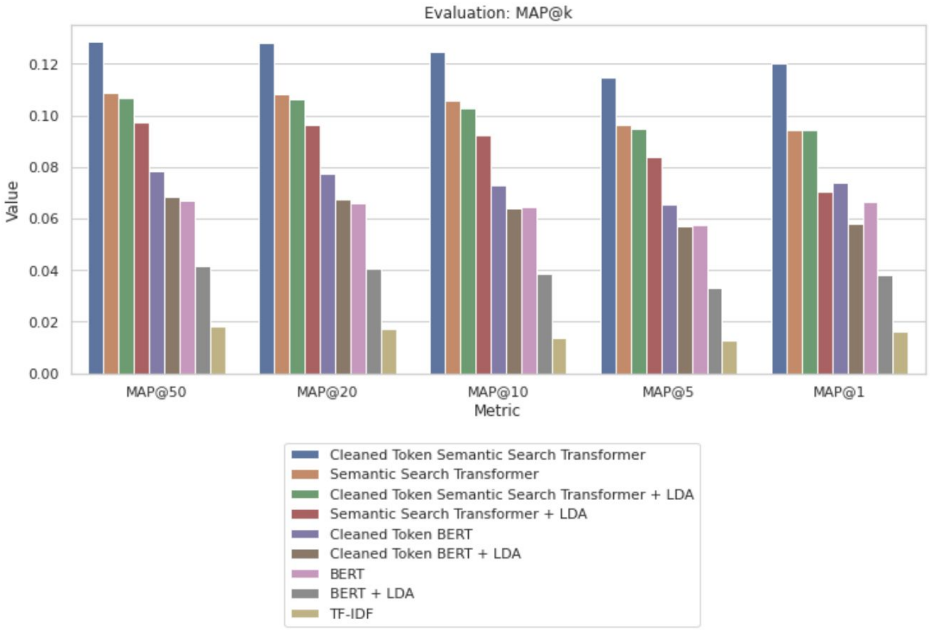
Experiment	Architecture	Cleaned?	LDA Appended?
TF-IDF	TF-IDF		
BERT	BERT Base		
Cleaned Token BERT	BERT Base		
BERT + LDA	BERT Base		
Cleaned Token BERT + LDA	BERT Base		
Semantic Search Transformer	Semantic Search Transformer		
Cleaned Token Semantic Search Transformer	Semantic Search Transformer		
Semantic Search Transformer + LDA	Semantic Search Transformer		
Cleaned Token Semantic Search Transformer + LDA	Semantic Search Transformer		

Evaluation Metrics

1. NDCG@k
    a. Normalized discounted cumulative gain
    b. How close is the predicted ranking of k items to the ideal ranking of k items that a user would prefer
2. MAP@k
    a. Mean average precision
    b. How many k items did our recommendation system predict which are actually within the labelled dataset

Evaluation Metric Results

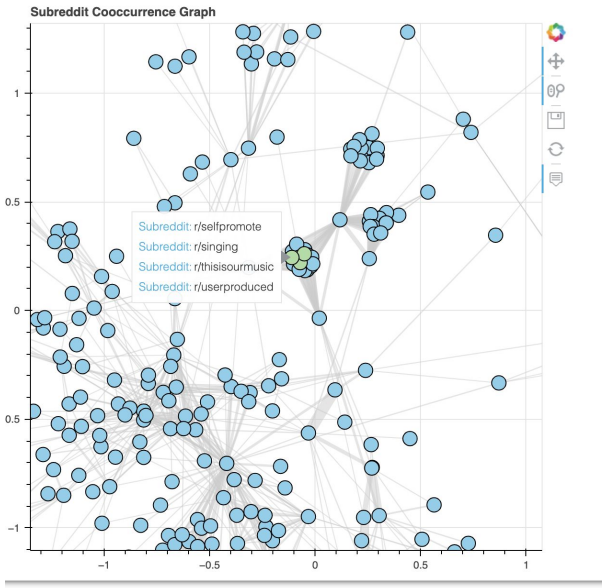




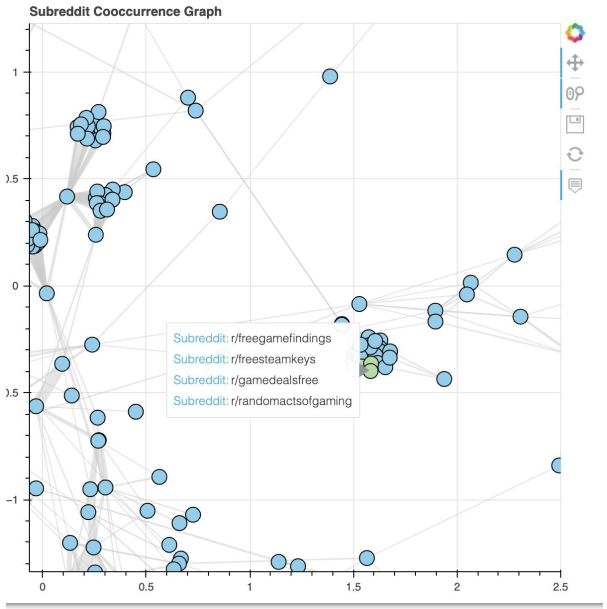
Knowledge Graph

- We can enhance our subreddit recommendations using a knowledge graph
- We developed a subreddit co-occurrence graph from our data
  - Node: subreddit
  - Edge: number of posts which mention a pair of subreddits together
- Provides extra related subreddits branched off from any subreddit starting point based on aggregated user signals of relevance
- Combine with results provided from recommender/ranker system

Knowledge Graph Examples

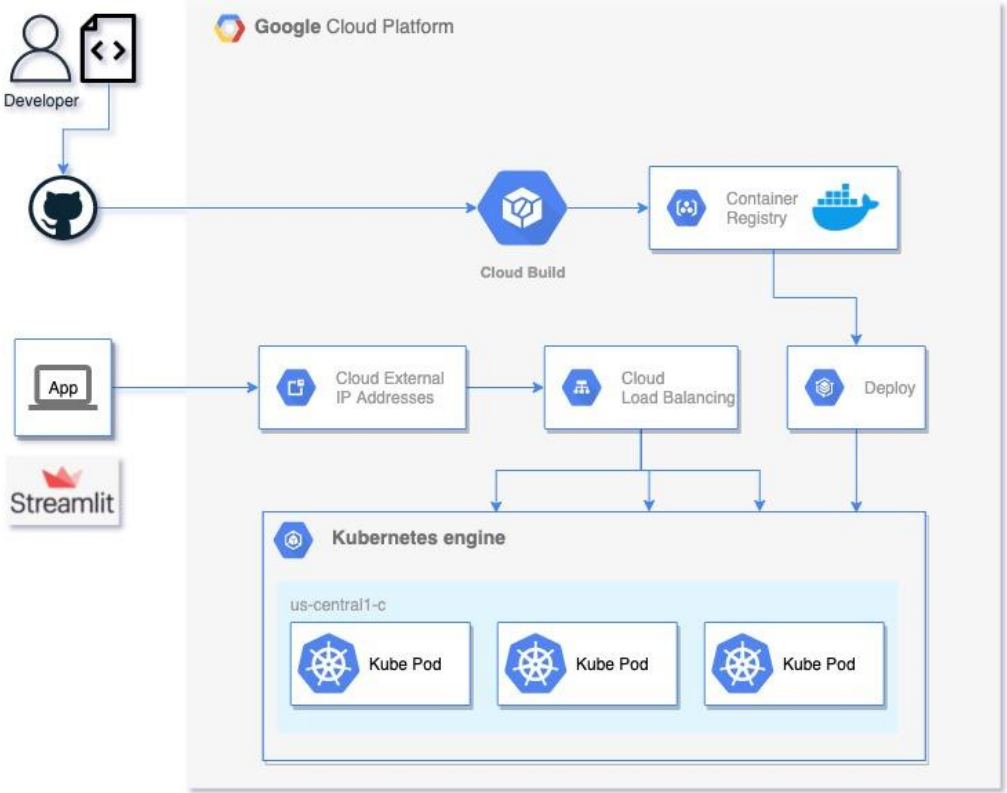


The green circles show a cluster of connected subreddits related to singing and music production.

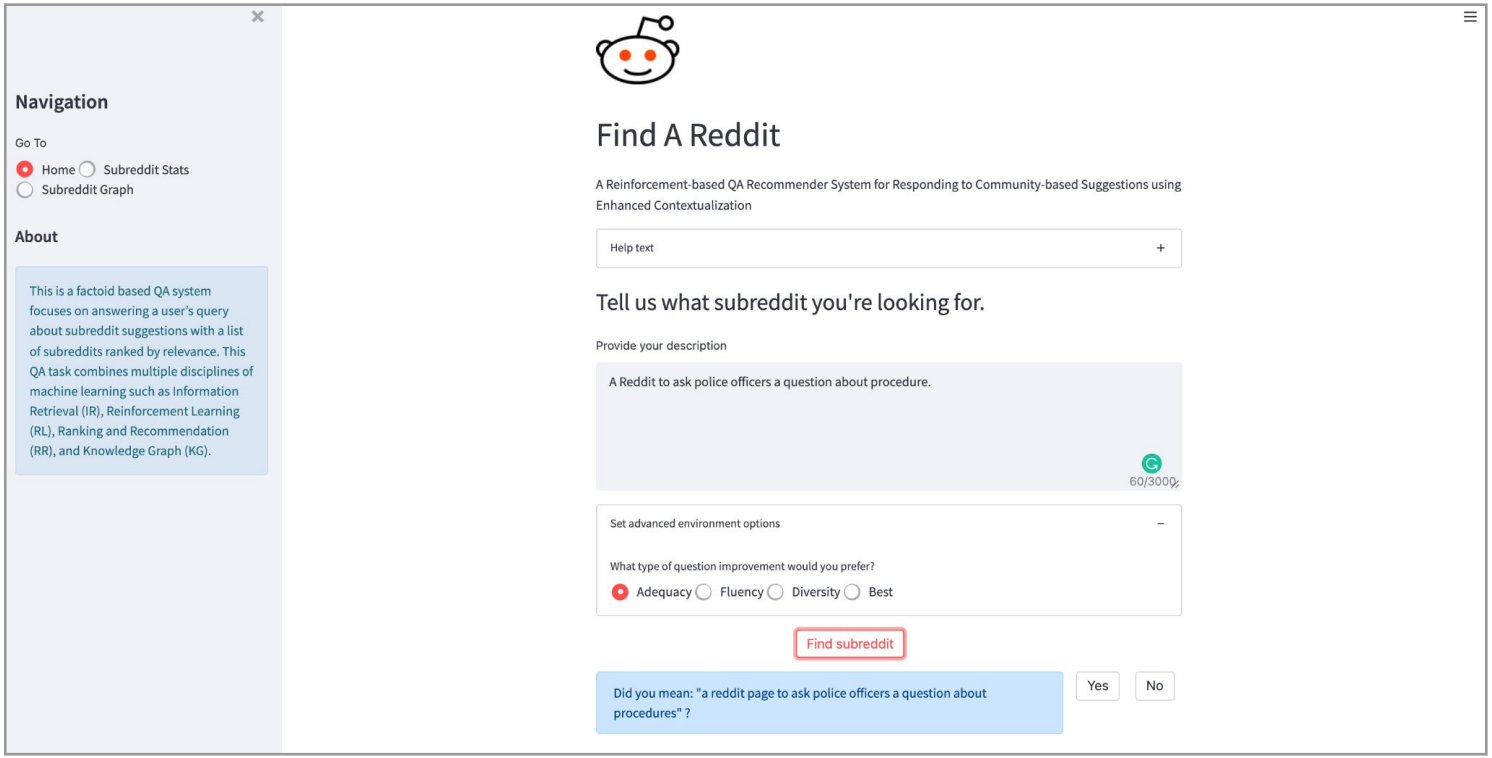


The green circles show a cluster of connected subreddits related to free games.

Deployment


Google Cloud Platform
Developer
Cloud Build
Container Registry
App
Streamlit
Cloud External IP Addresses
Cloud Load Balancing
Deploy
Kubernetes engine
us-central1-c
Kube Pod
Kube Pod
Kube Pod

Demo


Navigation
Go To
Home
Subreddit Stats
Subreddit Graph
About
This is a factoid based QA system focuses on answering a user's query about subreddit suggestions with a list of subreddits ranked by relevance. This QA task combines multiple disciplines of machine learning such as Information Retrieval (IR), Reinforcement Learning (RL), Ranking and Recommendation (RR), and Knowledge Graph (KG).
Find A Reddit
A Reinforcement-based QA Recommender System for Responding to Community-based Suggestions using Enhanced Contextualization
Help text
Tell us what subreddit you're looking for.
Provide your description
A Reddit to ask police officers a question about procedure.
60/3000
Set advanced environment options
What type of question improvement would you prefer?
Adequacy
Fluency
Diversity
Best
Find subreddit
Did you mean: "a reddit page to ask police officers a question about procedures" ?
Yes
No

Conclusion

- We developed a useful subreddit recommender to help Reddit users find their online community of interest
- We followed a data and metric driven approach to implement a machine learning solution for our problem
- We followed best practices when deploying our system to Google Cloud Platform for real-time usage