Computer Engineering Department

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

Project Advisor: Ali Arsanjani

Bais, Abhishek (MS Software Engineering)
Feng, Haley (MS Software Engineering)
Joy, Princy (MS Software Engineering)
Phu, Shannon (MS Software Engineering)

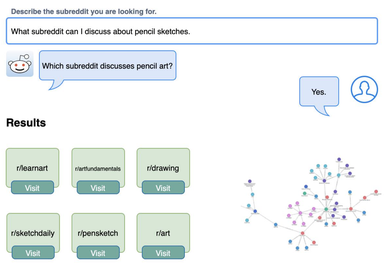
Introduction

In recent years, several Community-based Question Answering (CBQA) platforms spanning multiple verticals such as social news aggregation, business communication, retail, platforms for enthusiast programmers, and social media platforms for interactions and inquiry have gained traction. These platforms provide registered users access to a huge knowledge database of questions and answers accumulated over time. Users can post questions, seek crowdsourced answers, find shared interest groups, and learn from each other.

As CBQA platforms have gained popularity, the volume of questions and answers on them has increased exponentially. To help registered users find answers, several CBQA platforms organize their content in communities. Reddit organizes its content into topic-based communities called “subreddits”. Users can find relevant communities using tools such as “r/FindAReddit” and join them to post questions and get community help on finding which subreddits to join. However, with over 2,620,000 subreddits and an average of 60,251 new subreddits added to Reddit each month, finding the right community to join is a challenging task [1].

Proposal

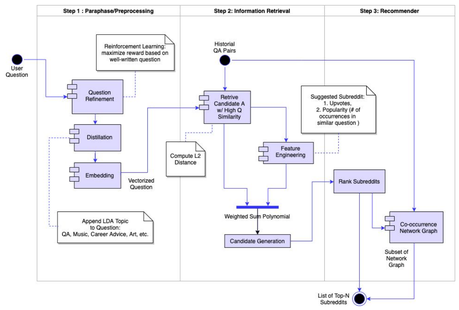
To address this problem, a Reinforcement-based Question-Answer Recommender System (RQAR) is proposed to help CBQA users find relevant communities for their questions. This system will be used to extract context, group previously answered questions by similarity, learn from user interactions to find, rank, and recommend relevant communities to users.



Architecture

Our system has three components:

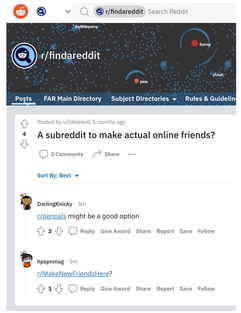
- Data processing
  - Data collection
  - Question refinement
- Information Retrieval
- Recommender



Methodology and Results

Data Collection and Processing

We collected the training data for our machine learning model to create a question-answer database scraped from Reddit using scraping APIs such as PRAW and Pushshift. We collected 1 year of data between 7/1/2020 - 7/1/2021 for 40 thousand total posts containing questions, answers, and upvote information. We cleaned the data’s text and for parsing issues.

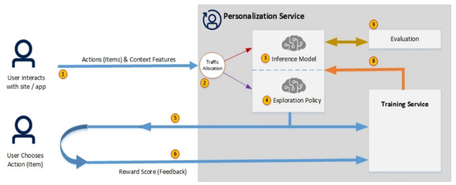


question_id	question	question_vocab	comment_upvotes	suggested_subreddits
o0oakk	A subreddit to make actual online friends?	make actual friend title	2	r/penpals
o0oakk	A subreddit to make actual online friends?	make actual friend title	1	r/makenewfriendshere

Reinforcement-based Question Refining

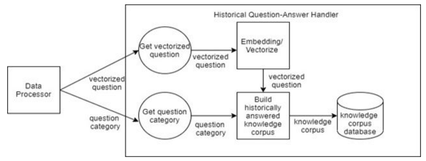
The Paraphrase and Preprocessing Unit collects a user posted question and proposes a paraphrase back to the user using Reinforcement Learning. The Question Refinement module uses a policy and a rewards function aimed at maximizing the reward for the well-formed question. It does so using the Microsoft Azure Personalizer service and Parrot paraphraser utility built on top of the Text to Text Transfer Transformer (T5). These rewards are used by the system to generate refined questions that are provided to the user to seek feedback and improve subsequent queries. If a user likes the rephrased question, the new question will be used to recommend a list of subreddits of interest.

```
---------------------------------------------------------------------------------------------
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?
---------------------------------------------------------------------------------------------
```



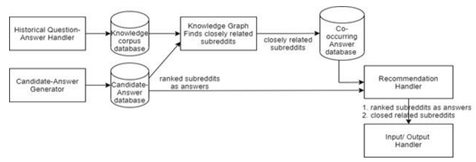
Information Retrieval

The Information Retrieval Unit consists of a QA database generated from Reddit scraping APIs such as PRAW and Pushshift. A candidate answer retrieval API calculates the semantic similarity between the user’s question and all historical questions using cosine similarity. Historical questions with high semantic similarity are used to query related subreddits and their upvotes. Additional features such as popularity of the subreddits among related questions are created during this process. The semantic similarity and additional features of these subreddits are used to compute a final weighted score.



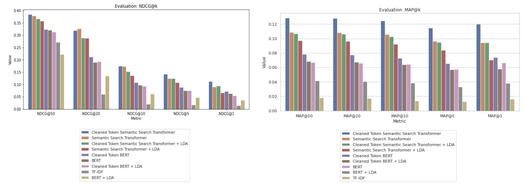
Recommender

The Rank and Recommend Unit ranks the candidate answers generated by the Information Retrieval Unit by total relevance weight to make available high-quality community recommendations to the user as answers. A separate co-occurrence graph API builds a network knowledge graph using all subreddits from the QA database. A subset of the network graph will be displayed along with the recommendation to show the relationship between the final top N subreddits.



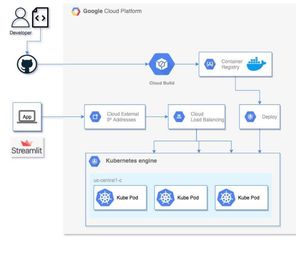
Experiments

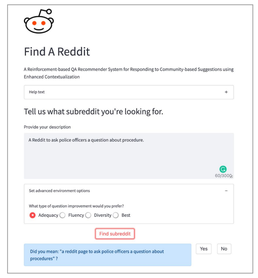
Experiment	Architecture	Cleaned?	LDA Appended?
TF-IDF	TF-IDF		
BERT	BERT Base		
Cleaned Token BERT	BERT Base		
BERT + LDA	BERT Base		
Cleaned Token BERT + LDA	BERT Base		
Semantic Search Transformer	Semantic Search Transformer		
Cleaned Token Semantic Search Transformer	Semantic Search Transformer		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



Deployment

We deployed our web application on Google Cloud Platform. We utilized Kubernetes engine to deploy a Streamlit Python web application. We integrated a CI/CD pipeline through Github such that each merge into the main branch would trigger a deployment. Each deployment builds inside a Docker container and serves the latest models and code to the hosted web application.





Summary/Conclusions

In summary, we saw a need for a question-answering service to help online users find communities of interest. We designed and developed a platform for users looking for their particular subreddit of interest, and returned a ranked list of subreddits they could potentially post to.

We experimented with many different techniques and found “Cleaned Token Semantic Search Transformer” to perform best against a hidden test set according to our metrics, NDCG and MAP.

We additionally productionized our model through a web client using Google Cloud Platform and Kubernetes Engine to serve predictions to users on the website.

Key References

[1] B. Molina, Reddit is extremely popular. Here’s how to watch what your kids are doing, USA Today, July 26, 2018. Accessed on: June 20, 2021. [Online] Available: https://www.usatoday.com/story/tech/talkingtech/2017/08/31/reddit-extremely-popular-heres-how-watch-what-your-kids-doing/607996001

Acknowledgements

We express great gratitude and appreciation to our advisor Dr. Ali Arsanjani for his guidance and support throughout the project.