# Campus Connect: A cross platform Question and Answering app with Recommendation System

Jayesh Kale[1], Hetansh Bharad[2], Sourav Ghatage[3], Soham Meher[4], Dr.Shwetambari Chiwhane[5]

*[1,2,3,4]B.E Student, Dept of computer Engineering, NBN Sinhagd school of Engineering, Ambegaon, Pune-411041, Maharashtra, India*
*[5]Prof. Dr., Dept of computer Engineering, NBN Sinhagd school of Engineering, Ambegaon, Pune-411041,    Maharashtra, India*

***Abstract***
*Question and answer (Q&A) forums are getting bigger day by day as number of internet user increases every user will have a specific and new problem which is either already solved or yet to be solved. So for creating a platform for college student where they can find all the relatable questions and also ask questions about the studies whose answer can be given by the community. In this paper we are focusing on the implementation survey on various features and technology which are involved in creating a forum with a recommendation engine. Explained steps are Recommendation engine, Leading user detecting, FAQ detection and Open Information extraction system. Based on the analysis, the ideal process and procedure is acknowledged, which will make the forum effective and used in all devices.*

## 1. INTRODUCTION

Analysis and review being done on various techniques applied for question and answering system. A brief analysis of each method was carried out to identify its own merits and limitation. The literature survey has different approaches/tools to develop an efficient question and answering system. The survey helps to gather ideas from the existing methods such as accuracy of recommendation engine and leading users and open information extraction system. Technologies we are using to accomplish this are namely flutter, machine learning, flask, dart.

## 2. RELATED WORK

### 2.1. The Base Idea

Many questions of student may remain unanswered whether that is academic or extracurricular due to the time constraint of teachers. There are no feasible options available that specifically address above issue regarding several different colleges or campuses. So its become important to clear the doubt of student in their area of interest as well as increase their knowledge. The proposed system will serve as platform for effective communication between student and teachers. Thus, the application will work as student asking question and community providing a helpful response. As our recommendation algorithm kick in those questions are recommended to the user on feed page using the tags which attached to the question which are required to type in while asking question.  User can create, edit, update, delete question and answer. The answer of the question will be ranked based upon the up votes and down votes. User will be able to bookmarks the questions as well as he/she can view asked question by him/her and answer given by him/her by taping on side panel.

### 2.2. Architecture
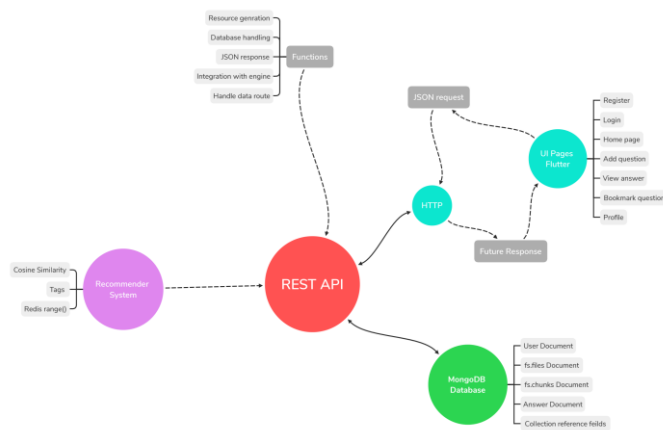
There are many approach to build full stack application such as front to back, back to front, constructing middle-ware first, etc. But we are using Front-To-Back Approach as creating user interface first will help us in mapping different relationship between the fields of form and routing once specific action is performed(State change). As well as form field defined in pages will became the solid foundation for creating and accessing variables in REST API. Starting with implementation of user interface which are created using Flutter(framework) and Dart(Programming language). The user interface pages include login, sign up, feeds, profile, create question, give answer. After the successful construction of pages we have to create model classes and future function which return json response to/from rest-api's, model classes usually map the value to json from UI forms. This mapped values are carried away by the http package of dart programming language over the network to communicate with rest apis in this case localhost of same machine. Next step is to define database relationship mapping with its element and access pattern before starting to make rest api's. After successful relation mapping we are finally able to create rest api's in flask micro-framework, for that we need dependency called flask-restful which are further discussed in **Technology and dependency** section. Using class based approach created resource which will be either consumed by frontend or will accept data. Next step is to construct and train our content based recommendation engine which will give prediction based upon the tags in question and selected tags by users. Recommendation engine is made with the logic of cosine similarity to match similar tags of user and question pared with radius in memory data store to retrieve in order of their score.

### 2.3. Technologies and dependency
### 2.3.1 Flutter
Flutter is a frontend framework and UI library which is cross platform and support almost all platform such as linux, windows, macos and android, ios, web but linux and windows are still in beta phase. Dependency of we required are http library for network request, dio for multipart form data submission.

### 2.3.2 Flask
Flask is micro-framework for python prgramming language and best suitable for custom application. Support large scale application on concept of plugins. We are also using these plugins as dependency includes python 3.8, flask_restful for creating rest api using class based concept, pymongo for connecting mongodb database and perform query, bcrypt for hashing passwords, jacard coeficient for searching, gunicron if deployed. Flask is our backend which serve flutter UI with rest apis and recommendations.

### 2.3.3 Databases
MongoDB and redis are two databases we are using for project. Mongodb is used as regular database while redis is used to store recommendation in memory as data store. Mongodb is document oriented NoSQL database. Redis is quite special can be used as key-values database or as in memory data

structrue, cache, messaging platform. All our user information and there referenced information is stored in MongoDB collections. For storing images GridFS is best option which store large files in smaller chunks.

## 2.4 Recommendation System using Content Filtering

The purpose of recommendendation system is to design an algorithm which knows the needs and different types of levels of which work efficiently. When in requirement gathering and designing phase of the project life cycle it gives better solutions. The companies now have moved towards the recommendation system as to increase their coverage in the market. As per the information gathered on the user the recommendations are made which can either be related to products and their services. This type of recommendation has three categoies which are as follows :

1. Collaborative recommendation

In this approach the data clusters are made from the available data which is filtered from the overall information given. This type of approach collects the data from other users and then the data is collaberated and the recommendation are given or generated on based on that. For eg. the user wants the recommendations to watch which movie then the as per the comparison on the prefrences information the recommendation will be generated.

2. Content-based recommendation

In this recommendation it is a comparison between the individuals profile and the contents of the product. This type of technique is easy to implement and can make changes whenever required, these are also transparent in nature and due to which this type of recommendation engine produce the recommendations which contain high relatability. For refering purposes the profile is used as the result may consists of some similar recommendation keywords. The ideas of a single user are compared with one another.

3. Hybrid recommendation

This type of recommendation is a mixture of multiple recommendation algorithms. It takes both the content-based and collaborative approach to achieve the wanted or required output. When multiple data clusters are being used which are with and without information, which are used where developing an result is necessary.
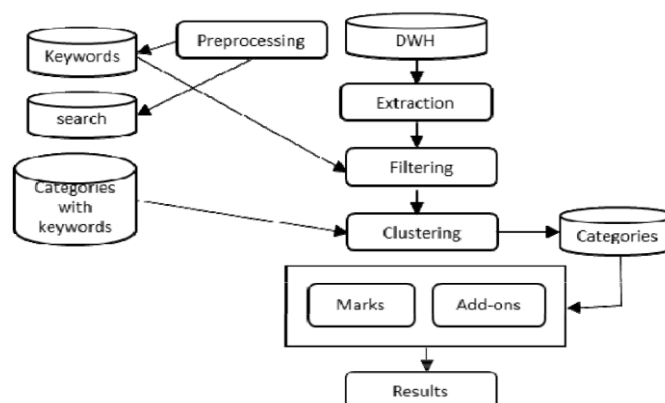


## 2.5 Analysis of QAS with crowd sourcing

The main objective of analysis is to supply a platform for open communication to social media and community for finding their queries, To design of crowd sourcing application which will use for public required data, to make QA system that produces an appropriate declare the received question

and gift it to the verbalize for this purpose. The work includes the info for personal Crowd sourcing data and sort of question answer it's solely utilized by administrator. The principle of crowdsourcing in education-Crowdsourcing is large in different industries, however, has however to have an effect on

education. It does not be that academics area unit recreating lesson over and another time. With the common core dealing, it is sensible that academics would be exchanging best practices and methodology that job. Academics area unit probing for quality materials that are vetted.
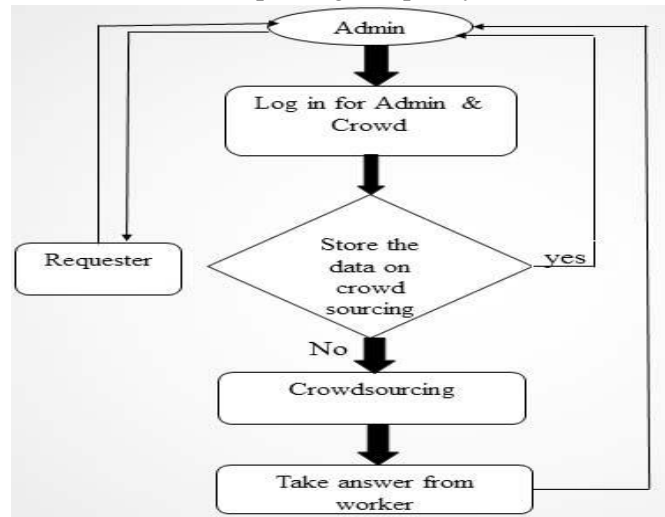


## 2.6 An Answer Recommendation for a Community Question Answering Systems

the question descriptions are represented with the query likelihood framework and word embedding methods. The similarity of two cases is measured by the distance of two vectors. The most similar historical cases could be selected according to the similarity ranking. Afterwards, in order to select the high quality answers, the answers belong to the cases are chose to estimate the quality. Another key operation in cQA is answers quality estimation. For each historical question, there are many answers from different doctors. The quality of the answers is varied because the doctors' experiments and knowledge are uneven. It's essential to select the high quality answers of the question. There are two kinds of popular approaches to assess the quality of answers, crowding sourcing methods [16, 17] and machine learning methods. To recommend answer to a particular question on medical QA systems. There are three steps for the answer recommendation system, including similar cases retrieval, answers quality estimation and answer recommendation. The data is collected from the QA system. The algorithm is tested on the dataset and the results show an increase in performance of answer recommendation.

## 2.7 Discovering Experts from Response Patterns

1. The classified DQ achieves an accuracy of more than 95% with 0.9 as the F1 measure. This classification shows the validation of the authenticity as it supports view count as a metric of difficulty level in the questions. FDQ feature is found if it is present one or more when checked the labeled DQ manually. The difficulty of a question relies on the combination of one or more topics knowledge which when present more than one topic's knowledge the level of difficulty of the questions increases. When combined the view count and favorite count for the assumption for the DQ then it produces better results. The 1a and 1b bar graphs shown are the accuracy and F1 represents the measure of the features of the DQ and all their different combined forms. For DQ recognition the

accuracy results are all above 50% which indicate that the feature selection has been done correctly however if all the features were to be combined then we have the highest accuracy results.

2. Experts Identification is represented in the table number 4 which represent the output of the differences in the list Ex1 & Ex 2. It represents the output of four various features of experts, the first approach is more of a conventional way for the recognition of experts, with the observation it is seen that the 3rd and 4th assumption of an expert is different and better than the conventional approach. The basis 3 observes that when the combination of best answer count and the answer count on the questions with the count of the answers being low, it provides high accuracy results and F1 metrics with a less set of features as compared to the best answer basis which is basis 1. When the difficult question feature is included i.e. FEX 11 is combined with clients of best answers as the experts give the best result with respect to the accuracy and the F1 score. This shows that with reduced set more dependable classifying model can be generated.

| Topic | Expert Basis | Instances | Approx % correctly matched instances | Accuracy | F1 measure |
|---|---|---|---|---|---|
| C | Best Answers | 31681 | 99 | 94.772 | 0.926 |
|  | Difficult Questions |  | 49 | 54.791 | 0.547 |
|  | Best Answers & Ans on Ques. Low Ans count |  | 99 | 99.369 | 0.991 |
|  | Best Answers & Difficult Questions |  | 99 | 99.598 | 0.995 |
| Python | Best Answers | 33747 | 99 | 94.724 | 0.927 |
|  | Difficult Questions |  | 66 | 61.151 | 0.581 |
|  | Best Answers & Ans on Ques. Low Ans count |  | 99 | 97.536 | 0.967 |
|  | Best Answers & Difficult Questions |  | 99 | 99.321 | 0.99 |
| Java | Best Answers | 40140 | 97 | 82.75 | 0.773 |
|  | Difficult Questions |  | 64 | 54.981 | 0.544 |
|  | Best Answers & Ans on Ques. Low Ans count |  | 99 | 97.262 | 0.963 |
|  | Best Answers & Difficult Questions |  | 99 | 98.125 | 0.974 |

## 3. SYSTEM
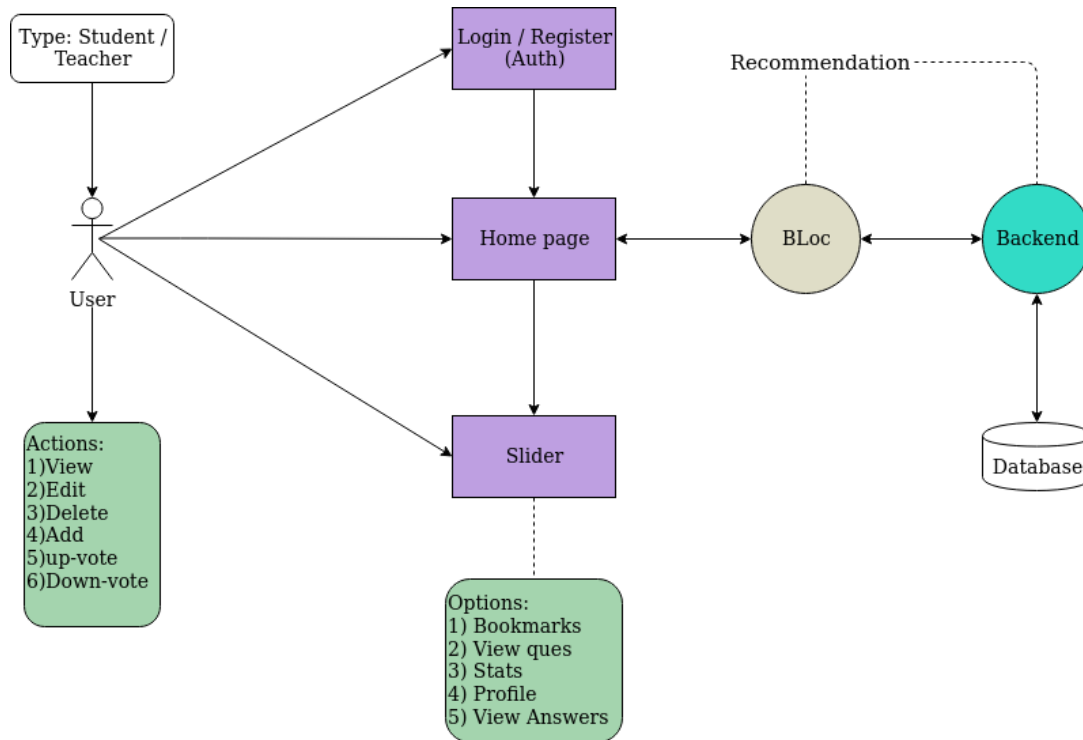
"Ask Question" is the module to ask question one at a time if the asked question is already exist in database it will generate exception and send it to user for this question is asked, Otherwise it will be redirect user to new screen. Recommended questions based on tags are appear on home screen as endless list of cards user can view related answer on tapping specific card. Profile and other information related to the user are included in side panel profile section. User can view all asked question using side panel questions and same way he can view answer which he write for answer. Anyone can able to upvote and downvote answers based on the upvotes the answer is sorted. User can create, edit, delete and update question and answer using right toggles. Images can be uploded in ask question section if theres any image regrading to question. Tags are mandatory to put in ask questions tag panel because these tags are keywords for our content-based recommendation engine. User can see tags quoted on question card. Theres login and logout functionality and to increase security pasword are encrypted using random ecnoding hashing algorithm. User can search question which is implented using jaccard index and search result are based on 50 percent or above similarity search can be toggled by search feild on appbar. User has only two type Teacher or Student.

## 4. RESULTS

The content based recommendation engine used, the model doesn't need any data about other users, since the recommendation are specific to the user which makes it easer to scale to a large numer of users. This type of engine capture the  specific interests of a user,can recommend question that few others are interested in.  Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE) are the metrics we use to find and diffrentiate the results from the other recommenadation engine's and on average the RMSE score for the content based filtering is 0.93 and the average score for the MAE is 0.72, the less the number of RMSE and MAE produced the better we can say that have the better recommendation results. The content based recommendation system has shown better results as compared to the random recommendation system.

## 5. CONCLUSION

Based on this literature survey, presents an analysis of different techniques used in the process of making question and answer forum. This study paper provides the fundamental techniques, recommendation procedure, and ranking methods required in order to build Q&A forum or platform. Exploring the proposed work in field of Q&A platform principle, literature's and commercial utilization experiences, different feature and techniques to achieve performance, robustness, security and accuracy. A careful analysis of requirement and system performance will be helpful for developing new efficient algorithm with better performance in the research work. For future research work we intend to consider the other option and metrics to evaluate the performance.

## REFERENCES

[1]"Answer Ranking in Community Question Answer(QA) System and Questions Recommendation" by "Sunny S. Shah", "Tejas S. Bavaskar", "Sourabh S. Ukhale", "Akash S. Kalyankar".

[2]"Robust Web Service Recommendation via Quantile Matrix Factorization" by "Rui Zhu", "Di Niu" and "Zongpeng Li".

[3]"Question Answering System Using Web Snippets" by "Menaha R", "Udhaya Surya A", "Nandhni K" and "Ishwarya M."

[4]"Shwetambari Kharabe", "C. Nalini", "Robust ROI Localization Based Finger Vein Authentication Using Adaptive Thresholding Extraction with Deep Learning Technique", Journal of Advanced Research in Dynamical & Control Systems, Vol. 10, 07-Special Issue, 2018.

[5]"Shwetambari Kharabe", "C. Nalini", "Using Adaptive Thresholding Extraction - Robust ROI Localization Based Finger Vein Authentication", Journal of Advanced Research in Dynamical & Control Systems, Vol. 10, 13-Special Issue, 2018.

[6]"Exploting response patterns for identifying topical experts in stackoverflow" by "Manish Bhanu", "Joydeep Chandra."

[7]"An Open Information Extraction For Question Answering System" by "D.Thenmozhi", "G.Ravi Kumar".

[8]"Leading Users Detecting Model in Professional Community Question Answering Services" by "Siqi Song", "Ye Tian", "Wenwen Han", "Xirong Que", "Wendong Wang".