Data Driven Answer Retrieval in Offline QA System

Rahul Nirmal[1] ,Samir Patil[2,] Avinash Ghorpade[3], Yogesh Murkunde[4] ,
Prof P.A Tak [5]
Zeal College of Engineering and Research, Pune

Abstract:

Searching the question in the previous asked questions (historical question) to get the meaningful answers. The searched question has more than one answers means we can get pool of multiple answers. Because of this it will take lot of time to browse all the pool of answers and go through it and choose the best one. To solve this problem in this paper we are providing the ranked answers in the form of pairwise comparisons. In particular, it consists of one offline learning component and one online search component. In offline we can find the sentiment in positive, negative and neutral categories to find the proper rank of the answers and suggest best one in that. In this paper provide these three types of training samples. In the online search component, we first collect a pool of answer for the given question via finding similar questions. Then sort the answer candidates by leveraging the offline trained model to judge the preference orders. We have supported the real-time datasets and work on the online and offline methodology.

Keywords: Question-answer, candidate pool, online, offline, sentiment

Introduction

In this paper we are using questions and answers as input and novel Pairwise Learning method to RANK model called PLANE, which can quantitatively rank answer candidates from the relevant question pool. We use Offline learning and online search where in offline learning which is guided by our user studies and observations, where we automatically establish the positive, negative, and neutral training samples in terms of preference pairs from input. And when it comes to the online search, for a given question, we pair it with each of the candidate's answer, and fit them into the trained PLANE model to estimate their matching scores. And like wise we help to find best answer

In the existing systems, When we try to find questions from QA systems we get lots of answers and to find what we want is very tough work because it consumes lots of time and we have to read each and every question manually so to solve this problem we are implementing this paper.

Earlier in QA system where questions were asked obtaining answers took lot of time and answer which answer we get is not what we want and the process is time consuming and it is not like when we send questions. Different methods where used to rank sort and divide answers.

In our project we are overcoming the drawbacks of previous systems. In our system, Natural Language Processing is being used to extract interest out of the post that is being search by the user. When

we try to find questions from QA systems we get lots of answers and to find what we want is very tough work because it consumes lots of time and we have to read each and every question manually so to solve this problem we are implementing this paper.

Currently there is generation of users ratings and reviews on different websites and system. System can study this ratings and recommendations and reviews to improve there software. In our system we are developing such system where the user can post a question related to anything and also post the answer and here In Community question and answer systems when we try to find questions we use archives where we can find them using theoretical base. But it can be time consuming part to find out questions and where they can be associated with different answers and to find out relevant answers they need to go through lot of answers to find what is needed.

The main Objective of the system is to identify related questions when a user post any question. When we try to find questions from QA systems we get lots of answers and to find what we want is very tough work because it consumes lots of time and we have to read each and every question manually so to solve this problem we are implementing this system.

System will be able to generate ratings from different domains. System deduces user interests based on his activities and post in social network.

The report mainly focuses on the discussion about different aspects of getting answers and the ways to get them as soon as possible. The first chapter deals with the introduction about the system defining the problem statement to which the system is focused on. It also discusses the scope of the system to which it works efficiently. It is also discusses the scope of the system to which it works efficiently. It also contains briefing of various research papers is to which it works efficiently. It also contains briefing of various research papers is to be used. The proceeding chapter contains the detailed SRS of the project describing the user feasibility, functioning of various modules, time line chart and process modeling. The third chapter contains IDEA matrix, mathematical model and feasibility analysis. It also contains all the use case diagrams related to the project.

Problem Definition:

When we try to find questions from QA systems we get lots of answers and to find what we want is very tough work because it consumes lots of time and we have to read each and every question manually so to solve this problem we are implementing this paper.

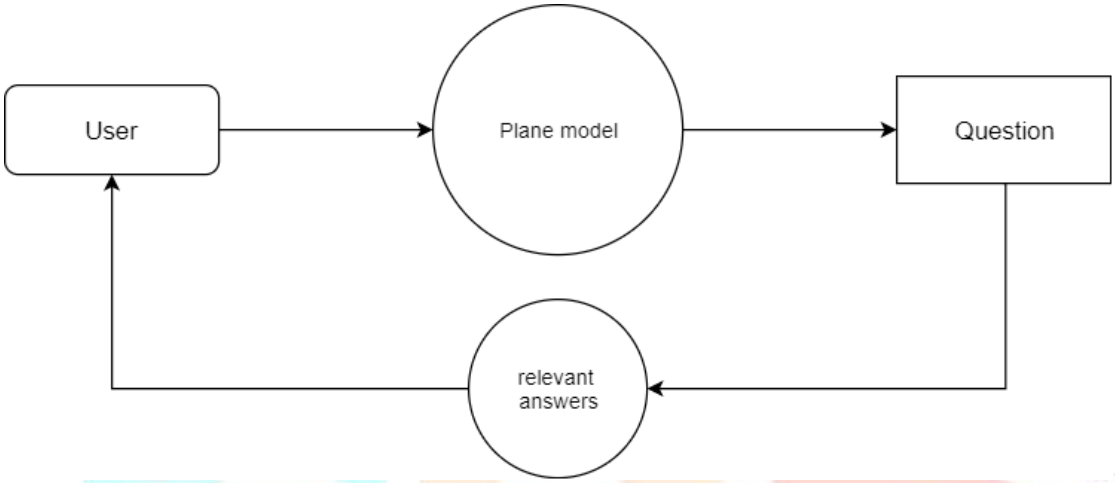
Objectives

- The main objective is to provide users to get information from CQA systems where lots of discussion is done and to find out information we are providing this system.

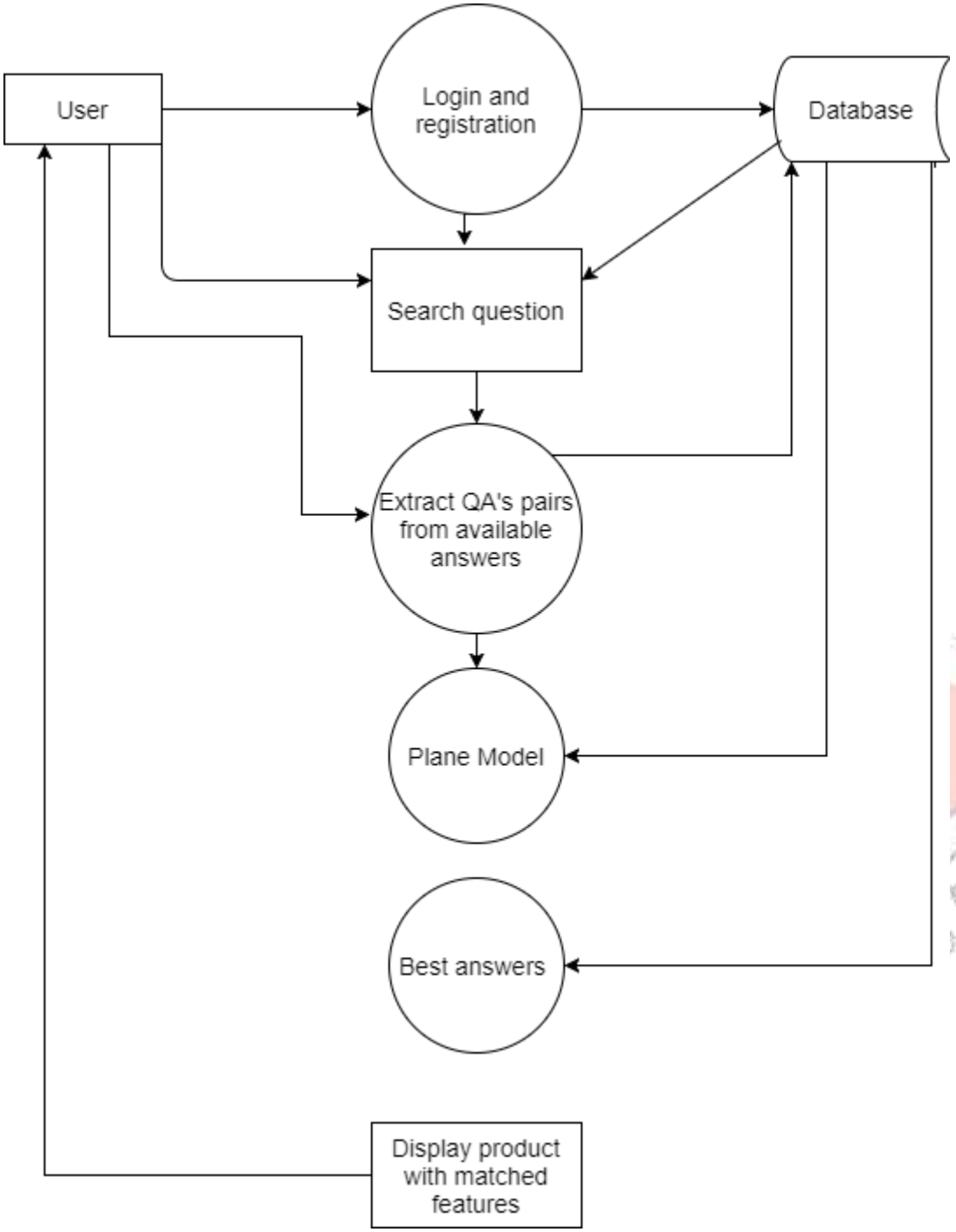
- Help user to get best and relevant answers for question searched.
- Help to save time of users by providing relevant answers.
- Provide user with answers which are related to topics searched.

System design:

Data Flow Diagram 0:

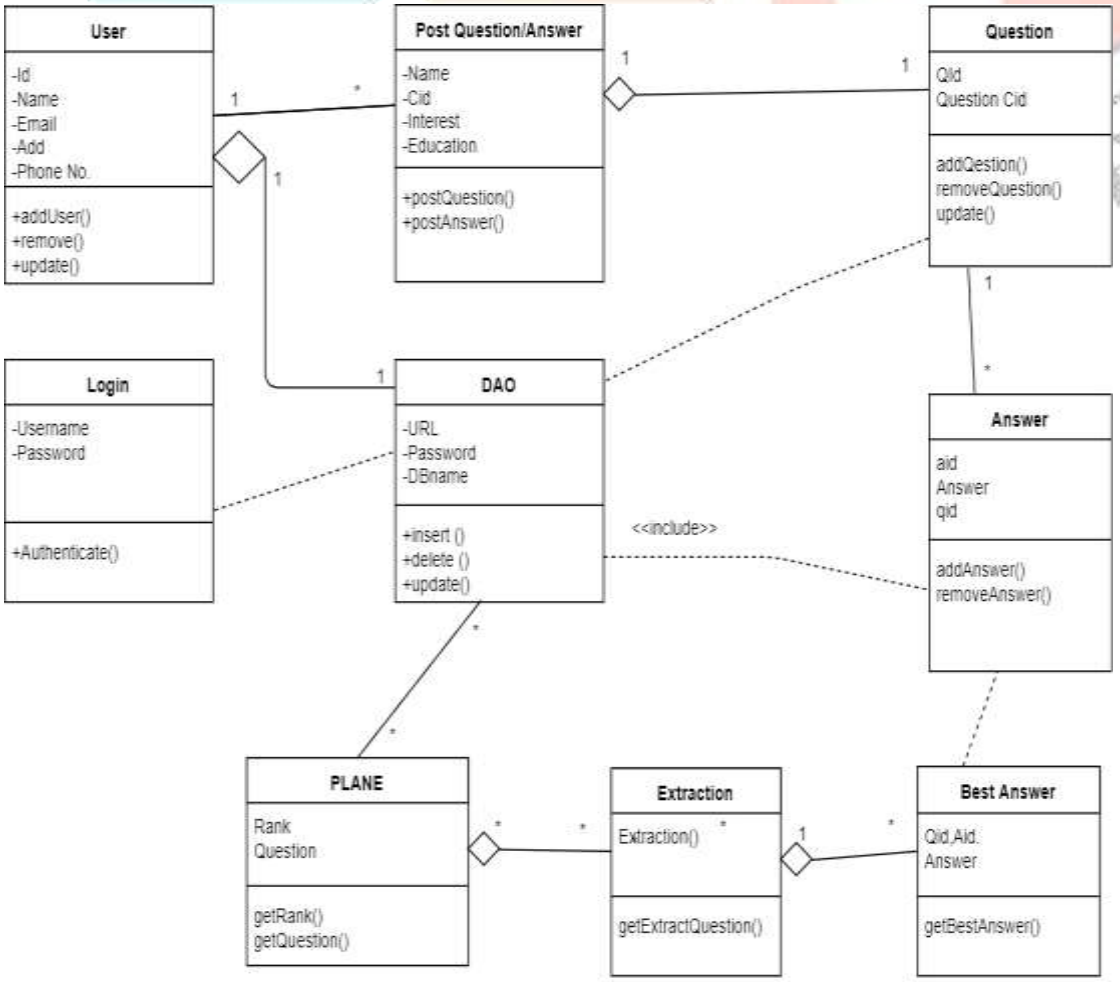


Data Flow Diagram 1:

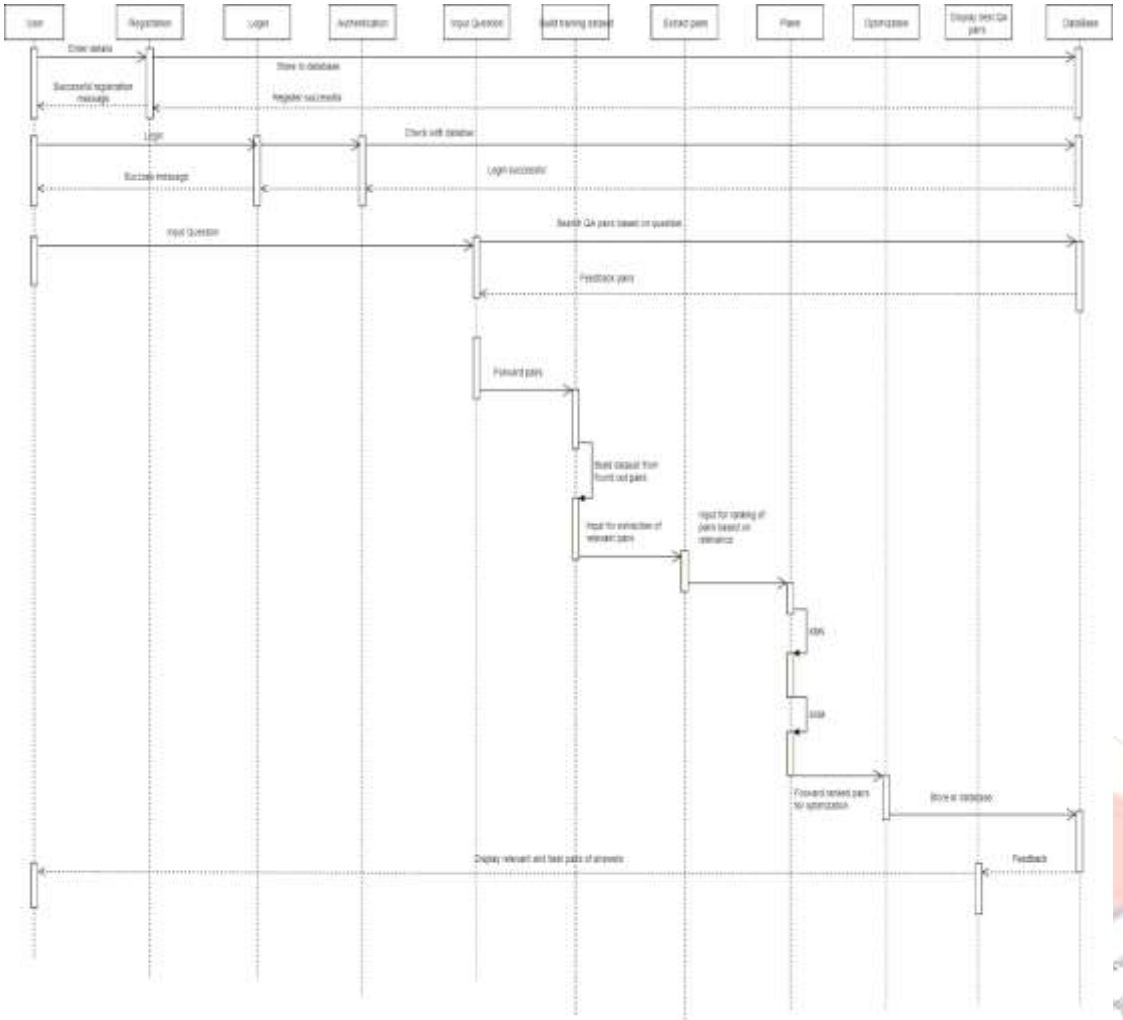


ER Diagram:

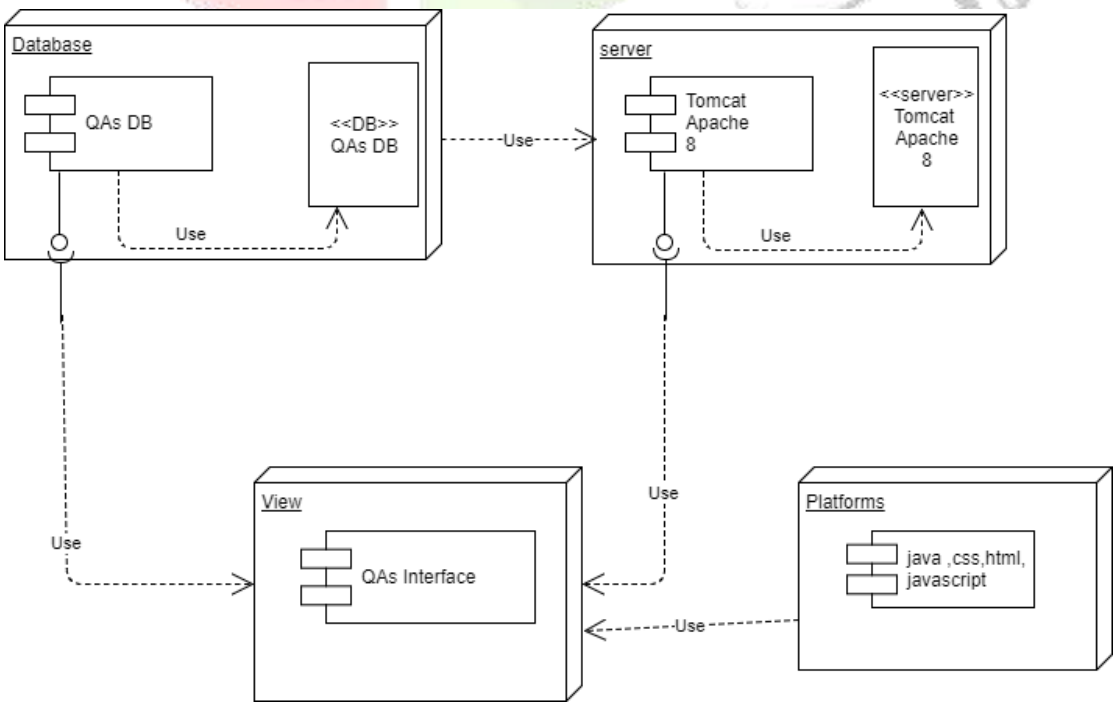
Class Diagram:



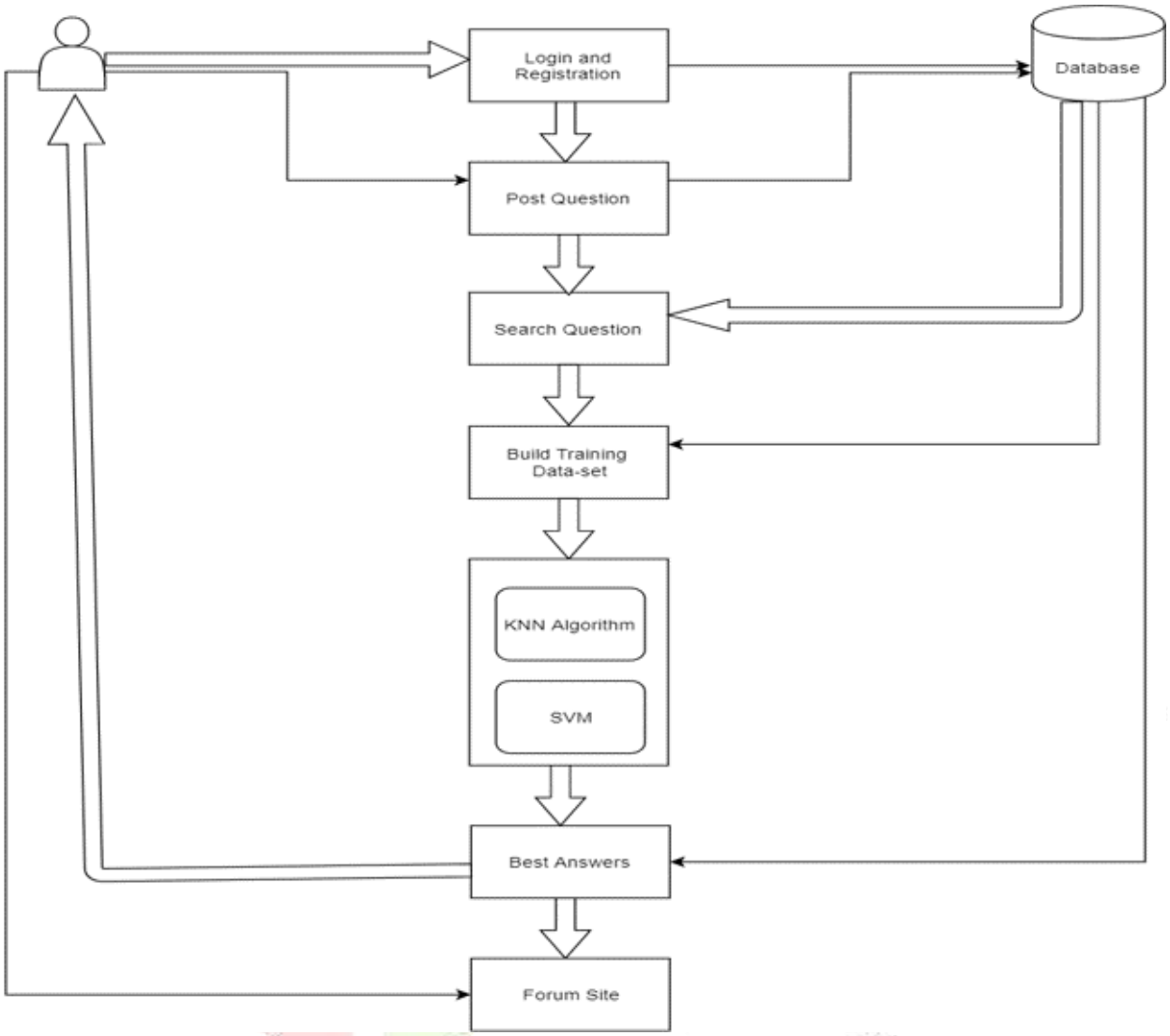
Sequence Diagram:



Deployment diagram



SYSTEM ARCHITECTURE



Above fig shows system flow Search a question, instead of choosing the best answer from the most relevant question, in this paper, we present a novel Pairwise Learning to rank model, nicknamed PLANE, which can quantitatively rank answer candidates from the relevant question pools. In that two components are present: offline learning and online search. Particularly, during the offline learning calculate the sentiment positive, negative, and neutral training samples in terms of preference pairs. The PLANE model can be jointly trained with these three kinds of training samples. We conduct extensive experiments over two datasets, collected from a vertical CQA site HealthTap and a general CQA site Zhihu.com, respectively

Related Works

2. 1 Combining multiple features for Answer Selection in Community Question Answering :

Our system combines 16 features belonging to 5 groups to predict answer quality. Our final model achieves the best result in subtask A for English, both in accuracy and F1- score. We only take part in subtask A for English and there can be lots of confusion. And for that we will use semantically rich representations of text will be used for improvement in performance.

2.2 Learning to find topic expert in a twitter via different relation :

Based on the current trending topics and hashtags we fetch the answers.

It working on blog and microblog technique where we find experts based on information but need to work on finding experts with good knowledge and interest of expert.

2.3 Learning to Recommend Descriptive Tags for Questions in Social Forums :

In this system enables each question to have multiple manually assigned topics without constraints on the vocabulary. These tags summarize question content in a coarse-grained but semantically meaningful level. This work unravels the incomplete and biased problems of question tags. And we will work on it in our future work

2.4 Improved Answer Ranking in Social Question-Answering Portals :

In this paper we work on usefulness of our features and query expansion techniques, and point to the importance of regularization when learning from noisy data. And to address an extension of the described approach to an end-to end QA system that includes a question-question mapping and a ranking over the full space of answers.

2.5 Beyond Text QA: Multimedia Answer Generation by Harvesting Web Information

Different from the conventional MMQA research that aims to automatically generate multimedia answers with given questions, our approach is built based on the community-contributed answers, and it can thus deal with more general questions and achieve better performance. Another problem is the lack of diversity of the generated media data. We have adopted a method to remove duplicates, but in many cases more diverse results may be better. In our future work, we will further improve the scheme, such as developing better query generation method and investigating the relevant segments from a video

DESIGN OF THE STUDY

Propose Algorithm:-

Step 1: Search question.

Step 2: Find QA pairs for entered question.

Step 3: Build training dataset.

Step 4: Forward QAs pairs.

Step 5: Plane model (Ranking of pairs using KNN and SVM).

Step 6: Optimization on pairs.

Step 7: Display relevant QAs pairs.

TOOLS USED

- JDK 1.8 or higher version
- Eclipse Mars or higher version
- MySQL 5.7 or higher version
- Tomcat 8 or higher version

Software Requirement:

- Operating System : windows 8 and above..
- Application Server : Tomcat5.0/6.X
- Language : Java
- Front End : Java 8
- Database : MySQL

Hardware Requirement:

The hardware design of the system includes designing the hardware units and the interface between those units.

- Processor - Intel i3/i5/i7
- RAM - 4 GB (min)
- Hard Disk - 50 GB

STATISTICAL TECHNIQUE USED

We have used data mining techniques such as search keywords, feature extraction in our project to extract the information from search questions. It is concerned with where the system proposed for development for computerized. The hardware development is so fantastic that there is hardly any business or job that cannot computerize

Experiment Result:

The result of our system is, In Community question and answer systems when we try to find questions we use archives where we can find them using theoretical base. But it can be time consuming part to find out questions and where they can be associated with different answers and to find out relevant answers they need to go through lot of answers to find what is needed.

Future scope:

In future we plan for developing a system that will overcome the disadvantage of the system. Our existing model is able to incorporate the neutral training samples and select the discriminative features. So here we are focusing on the best features.

Conclusion:

In this paper, we are providing new way to find best and relevant answers for asked questions. It supports with two online and offline components where in offline we train our system based on asked question and find answers based on it. In offline we calculate the create training samples in the forms of preference pairs using keywords in question. In the online search component, for a given question, we first collect a pool of answer candidates by finding its similar questions using plane model where we rank answers based on question and when user search he will be given best and relevant answer and then he can rate answers so that next time user will get that rated answer at top.

Reference:

[1] M. A. M. L. Wei Ding, He Jiang, Modern Advances in Intelligent Systems and Tools, ser. SCI. Springer, 2012, vol. 431. 1, 3

[2] L. Nie, M. Wang, L. Zhang, S. Yan, B. Zhang, and T. S. Chua,"Disease inference from health-related questions via sparse

deep learning," TKDE, vol. 27, no. 8, pp. 2107–2119, 2015. 1

[3] A. Shtok, G. Dror, Y. Maarek, and I. Szpektor, "Learning from the past: Answering new questions with past answers," in
Proceedings WWW'12. ACM, 2012, pp. 759–768. 1, 3
[4] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in Proceedings
of WSDM'08. ACM, 2008, pp. 183–194. 2
[5] J. Jeon, W. B. Croft, J. H. Lee, and S. Park, "A framework to predict the quality of answers with non-textual features," in
Proceedings of SIGIR'06. ACM, 2006, pp. 228–235. 2
[6] Z. Ji and B. Wang, "Learning to rank for question routing in community question answering," in Proceedings of CIKM'13.
ACM, 2013, pp. 2363–2368. 2
[7] T. C. Zhou, M. R. Lyu, and I. King, "A classificationbased approach to question routing in community question answering," in Proceedings of WWW'12. ACM, 2012, pp. 783– 790. 2
[8] L. Yang, M. Qiu, S. Gottipati, F. Zhu, J. Jiang, H. Sun, and Z. Chen, "Cqarank: Jointly model topics and expertise in community question answering," in Proceedings of CIKM'13. ACM, 2013, pp. 99–108. 2
[9] B. Li and I. King, "Routing questions to appropriate answerers in community question answering services," in Proceedings of CIKM'10. ACM, 2010, pp. 1585–1588. 2
[10] K. Wang, Z. Ming, and T.-S. Chua, "A syntactic tree matching approach to finding similar questions in community-based qa
services," in Proceedings of SIGIR'09. ACM, 2009, pp. 187–194. 2, 6
[11] Y. Liu, J. Bian, and E. Agichtein, "Predicting information seeker satisfaction in community question answering," in Proceedings of SIGIR'08, ser. SIGIR '08. ACM, 2008, pp. 483– 490. 2
[12] M. J. Blooma, A. Y. K. Chua, and D. H.-L. Goh, "A predictive framework for retrieving the best answer," in Proceedings of SAC'08. ACM, 2008, pp. 1107–1111.
[13] L. Nie, M. Wang, Y. Gao, Z. Zha, and T. Chua, "Beyond text QA: multimedia answer generation by harvesting web information," TMM, vol. 15, no. 2, pp. 426–441, 2013. 3
[14] Q. H. Tran, V. Duc, Tran, T. T. Vu, M. L. Nguyen, and S. B. Pham, "Jaist: Combining multiple features for answer
selection in community question answering," in Proceedings of SemEval'15. ACL, 2015, pp. 215C–219. 3
[15] W. Wei, Z. Ming, L. Nie, G. Li, J. Li, F. Zhu, T. Shang, and C. Luo, "Exploring heterogeneous features for query-focused
summarization of categorized community answers," Inf. Sci., vol. 330, pp. 403–423, 2016. 3
[16] S. Tellex, B. Katz, J. Lin, A. Fernandes, and G. Marton, "Quantitative evaluation of passage retrieval algorithms for question answering," in Proceedings of SIGIR'03. ACM, 2003, pp. 41–47. 3

[17] H. Cui, R. Sun, K. Li, M.-Y. Kan, and T.-S. Chua, "Question answering passage retrieval using dependency relations," in
Proceedings of SIGIR'05. ACM, 2005, pp. 400–407. 3

[18] R. Sun, H. Cui, K. Li, M.-Y. Kan, and T.-S. Chua, "Dependency relation matching for answer selection," in Proceedings of
SIGIR'05. ACM, 2005, pp. 651–652. 3

[19] M. Surdeanu, M. Ciaramita, and H. Zaragoza, "Learning to rank answers on large online qa collections," in Proceedings of ACL'08. ACL, 2008, pp. 719–727. 3

[20] A. Agarwal, H. Raghavan, K. Subbian, P. Melville, R. D.Lawrence, D. C. Gondek, and J. Fan, "Learning to rank forrobust question answering," in Proceedings of CIKM '12. ACM,2012, pp. 833–842.