INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

# Identifying Similar Question Pairs Using Machine Learning Techniques

**V K Raj Anishaa**[1]*, **P Sathvika**[2], **Sandeep Rawat**[3]

**1** Assistant System Engineer, Tata Consultancy Services, Deccan Park, Madhapur, 500081, Telangana, India
**2** Device, Digital and Alexa Support Associate, Amazon- Hyd20, Nanakaramguda, 500032, Telangana, India
**3** Professor, Computer Science and Engineering, Anurag University, Venkatapur, 500088, Telangana, India

OPEN ACCESS

**Received:** 19.02.2021

**Accepted:** 17.05.2021

**Published:** 04.06.2021

**Citation:** Anishaa VKR, Sathvika P, Rawat S (2021) Identifying Similar Question Pairs Using Machine Learning Techniques. Indian Journal of Science and Technology 14(20): 1635-1641. https://doi.org/10.17485/IJST/v14i20.312

* **Corresponding author**.

rajanishaa20@gmail.com

**Funding:** None

**Competing Interests:** None

Published By Indian Society for Education and Environment (iSee)

**ISSN**
Print: 0974-6846
Electronic: 0974-5645

## Abstract

**Background/Objectives**: Every day millions of people visit search engines like Quora, reedit, stack overflow, etc., the demand for new intelligent techniques is growing, to help individuals find better solutions. **Methods**: In our proposed system, the Quora datasets were filtered using SQLite which takes one-quarter of the time taken to pre-process the same dataset using existing approaches like python functions. We used machine learning techniques namely the Random Forest, Logistic Regression, Linear SVM (Support Vector Machine) and XGBoost to analyze and identify the most suitable model. **Findings**: The error log loss functions (0.887, 0.521, 0.654 and 0.357) of the above machine learning techniques were analyzed and compared respectively. The performance of XGBoost is the best among the other models, hence XGBoost is the most efficient model. **Conclusion/Future Scope**: It is concluded that XGBoost has outperformed other machine learning techniques discussed in the study. It is also found that pre-processing using SQLite has improved the response time. In the future, we would like to apply a similar approach for various other search engines that are available like reedit, stack overflow, etc. and one could ensemble the best models of each type (linear, tree-based, and neural network).

**Keywords:** Machine Learning; Question Pair Similarity; XGBoost; Linear SVM; Logistic Regression; Random Forest

## 1 Introduction

The need for a software system to locate information first appeared in the thought-provoking article titled “As We May Think” by Vannevar Bush in 1945. He proposed research libraries with related explanations similar to modern-day hyperlinks. The first systematic search engine was Archie created by Alan Emtage in 1990 and after that, there were many search engines like W3catalogue, World Wide Web Wanderer, Aliweb, JumpStation and WebCrawler, Yahoo, Megallan, Excite, Infoseek, Inktomi, Northern Light and AltaVista, Netscape and Bing appeared, until Google search engine by Larry Page emerged. Google collects information about the web pages, then categories web pages, finally making it available to people to access relevant web pages.

The objective of this study is to find the best machine learning algorithm to eliminate all the duplicate questions to enhance user satisfaction. Based on a critical evaluation many machine learning algorithms take a long time to train the real-time datasets. This study will highlight the key considerations for incorporating SQLite to preprocess the dataset, which in turn improves the response time by one-quarter of the time taken to pre-process the same dataset. The feature extraction of the similar question pairs in the Quora dataset is evaluated with features of question 1 and question 2 such as: frequency, length, number of words, number of common words, the total number of words, the sum of frequencies, the absolute difference of frequencies, etc. following which the performance of machine learning algorithms will be assessed under error log loss function and response time.

Many findings comparing the accuracy of different machine learning algorithms against the similarity search have been conducted(1–7). Zainab Imtiaz et al.(1) conversed about a model that determines pairs of duplicate queries by merging word embedding techniques such as FastText Crawl, FastText Crawl Subword and Google News Vector with separation techniques. This study proposed a new Siamese MaLSTM model that interprets the Manhattan distance to assess the logical similarity between several questions. But this approach is complex thereby increasing the time taken to produce results. W. Li et al.(2) presented a short-term regional Wind Power Prediction method based on multi-stage feature selection and XGBoost. The authors suggested that the Root Mean Squared Error (RMSE) of the XGBoost model is reduced by about 1% when compared to random forest and regression tree. This motivated us to use XGBoost in our work. Z. Xu et al.(8) discussed a Semantic Matching Model (SMM) incorporated with the framework of multitasking transfer for detecting duplicate questions. It introduced the word-to-sentence interaction approach depending on which possible similar words are either ignored or paid attention to. The drawback in this approach is that there is a probability that a potential similar word can be ignored. L. Wang et al.(9) used deep learning techniques to identify duplicate questions in a stack overflow. The proposed model incorporates Long Short-Term Memory (LSTM), Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN). Furthermore, word2vec was used to obtain the vector representations of words which is not feasible for huge datasets. D. A. Prabowo et al.(10) implemented CNN to determine the logical similarity of questions. Glove pre-trained word embeddings approach was used and the help of stochastic gradient descent was taken for optimization. But the accuracy obtained was comparatively very less about 79%. S. Mukherjee et al.(11) demonstrated a natural language system that used word embedding features like word2vec and simple hand-picked structural features for handling erroneous questions. But the recommended system is not efficient for complex queries. Mohammad Daoud(12) addressed the automatic Arabic query similarity identification. An approach based on rules incorporating semantic and lexical similarity was proposed and implemented. The authors specify that resemblance amid words could be determined by literal and logical similarity. In this work, Daoud classifies and arranges queries in the database concerning their logical structure considering their form and range of application. This approach is specific to a particular language i.e. Arabic. Chen, M et al.(3) give an overview of a prediction method of transient stability for power systems based on XGBoost. The authors discuss that more features can be taken into account in the XGBoost-based method, which is preferred in complex power systems, particularly in the power system penetrated by new energy sources, compared to the traditional method. Among the significant characteristics, which have the strongest association with system reliability, the major characteristics can be extracted. The simple analysis and quick calculation make it more feasible for a researcher to apply this methodology in the online power system predictions. The authors clarify that the XGBoost model yields accuracy high efficiency. Sultana R. et al.(4) discussed overcrowding in health care units caused by population rise over the last 10 years. Data mining models namely boosting and decision trees, originating from the tree-based approach were used for prediction activities. The latest method of gradient-boosting machines was elucidated for enhancing productivity and multiply speediness.

Similarly, other authors have identified many gaps in their work. Some of them even applied in their native language like Arabic sentence similarity(12) and Chinese sentence similarity(13). Most of these authors have identified gaps like lower precision and accuracy, suitable for document similarity but not for question pair similarity, etc. (Table 1) shows the comparison of different machine learning algorithms with existing gaps.

**Table 1.** Comparison of different machine learning algorithms showing existing gaps

| Author(s) | Algorithms/ Techniques | Datasets | Existing Gap |
|---|---|---|---|
| M. Daoud [12] | Weka 3.8 - Random Forests with 10 folds cross-validation | 300 questions prepared for Arabic language | Less accuracy |
| I. L. Cherif and A. Kortebi [5] | K-Nearest Neighbours (K-NN), Naive Bayes (NB) C5.0,C4.5, XGBoost | Data collected from a major French ISP network in 2015. | The time taken for training was not considered. |
| X. Dong, T. Lei, S. Jin and Z. Hou [6] | XGBoost | Traffic flow detector data collected in Beijing | Suggests that XGBoost improves precision and accuracy. |
| C. Saedi, J. Rodrigues, J. Silva, A. Branco and V. Maraev [7] | Rule-based approach, SVM and Deep CNN | Real-time dataset uploaded by Quora on Kaggle | It is not satisfactory for smaller datasets. |
| Shashi Shankar [14] | Support Vector Classifier model | Quora dataset | The implemented model has comparatively lower accuracy.<br>This work is more suitable for evaluating short answers. |
| B. Ye, G. Feng, A. Cui and M. Li [13] | RNN encoder-decoder | Constructed dataset containing 4,322 labelled question pairs in Chinese | This algorithm has been implemented specifically for the Chinese language. |
| Q. Mahmood, M. A. Qadir and M. T. Afzal [15] | SPARQL (Protocol and RDF (Resource Description Framework) Query Language) queries.<br>Sem SNA (Social Network Analysis) for analysis of RDF citation graphs. | Citeseer data set | This technique is more suitable for determining document-similarity as question pair similarity requires more detailed analysis. |
| J. Wang, Z. Li and B. Hu [16] | Question Similarity Based on Their Answers, Semantic Context from Existing Q/A Pairs, Cosine Distance | Community-based question and answer (cQA) service on the Web | This approach has lower accuracy when compared to other approaches. |

## 2 Proposed Models

The main objective is to determine whether the given pair of questions are duplicates of each other or not, thereby finding quality solutions to question ensuring enhanced experience for solution providers, readers and inquirers. Quora has uploaded its real-time data into Kaggle which consists of 404,290 question pairs that belong to the training data set and about 2,345,795 question pairs that belong to the test dataset. The size of the dataset after pre-processing is extracted to be 5.55GB. Four Machine Learning Algorithms namely- Random Forest, Linear SVM, Logistic Regression and XGBoost were used to identify the question pair similarity. The results obtained by each of the models are compared to identify the model which gives maximum accuracy in a minimum amount of time. The recommended architecture for analyzing the proposed system is shown in Figure 1. First data is collected from Kaggle, it is then analyzed and pre-processed. Noise is removed by eliminating HTML tags, stop words, punctuation, white spaces and URL. The obtained data is pre-processed using normalization. Normalization is done by using three sub-techniques called stemming, tokenization and lemmatization. Pre-processing is done using PL/SQL blocks in SQLite Database which can process a huge amount of data in less amount time when compared to other methods. Then, the obtained dataset is trained using a random model, Linear SVM, Logistic Regression and XGBoost. The results obtained are stored in the database. After which a comparison operation is implemented by analyzing the error log loss function for each of these training processes. The model with the least error function value is then selected thereby obtaining maximum efficiency and accuracy. Finally, a visualization operation is performed which represents the confusion matrix, precision matrix and recall matrix for each of the adopted machine learning models.

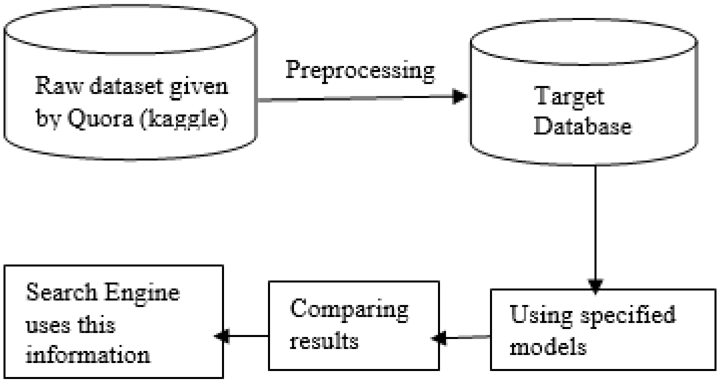
Raw dataset given by Quora (kaggle)
Preprocessing
Target Database
Using specified models
Comparing results
Search Engine uses this information

**Fig 1.** Functioning of the proposed model

## 3 Experiments & Findings

The output obtained for the four different models are demonstrated clearly and individually. The below formula (1) is used to compute the log loss (error function) for logistic regression.

$$-1/N\sum y_i * \log^{N}{}_{i=1}\left(p\left(y_i\right)\right)+\left(1-y_i\right)^{*}\log\left(1-p\left(y_i\right)\right) \tag{1}$$

The same log loss function yields different values for each of the specified four models as each of them have a unique implementation based on their specific algorithm.

### 3.1 Random Model

Figure 2 depicts three different matrices namely confusion matrix, recall matrix and precision matrix for the random model. The confusion matrix illustrates that the maximum value 9482 corresponds to the false positive (i.e. the actual value is no, but is predicted as yes). The log loss obtained is analogous to the error which has a value of 0.887 for the random model.

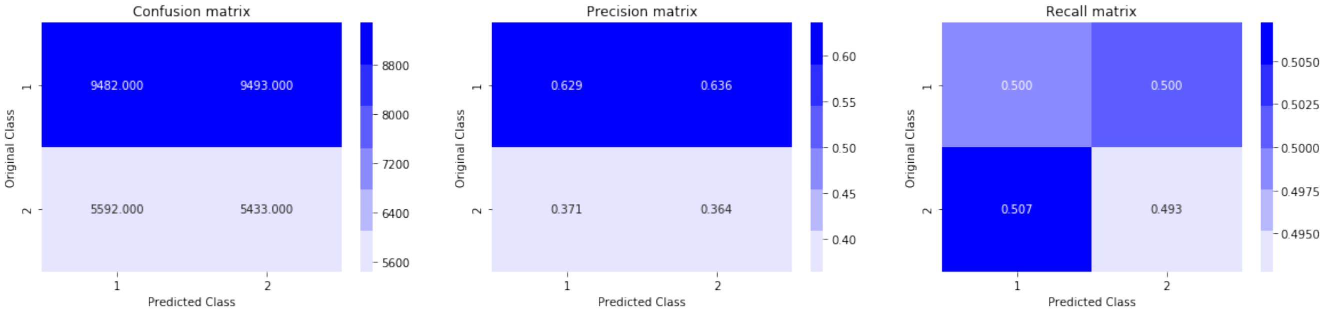

**Fig 2.** Confusion Matrix, Precision Matrix and Recall Matrix for the Random Model

### 3.2 Logistic Regression

Figure 3 shows the Cross-Validation Error for each alpha taken. It is an alpha vs. error measure graph. This elbow graph has been plotted to identify the ideal value of alpha for the length of the dataset. Figure 4 illustrates the matrices obtained for the logistic regression model. The confusion matrix demonstrates that the maximum value is obtained for True Positive (i.e. the actual value is yes and the predicted value is also yes) at 16551. Therefore, this model has a comparatively better performance. The log loss acquired for the regression model is 0.521 which is lesser than that obtained for the random model. Thus, logistic regression gives better accuracy and efficiency than the random model.

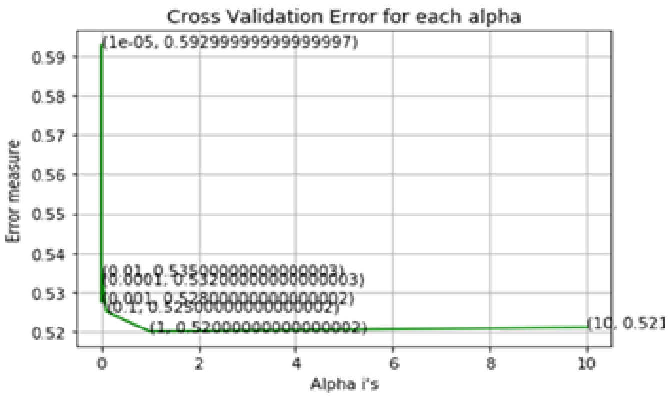

**Fig 3.** Cross-Validation for Logistic Regression

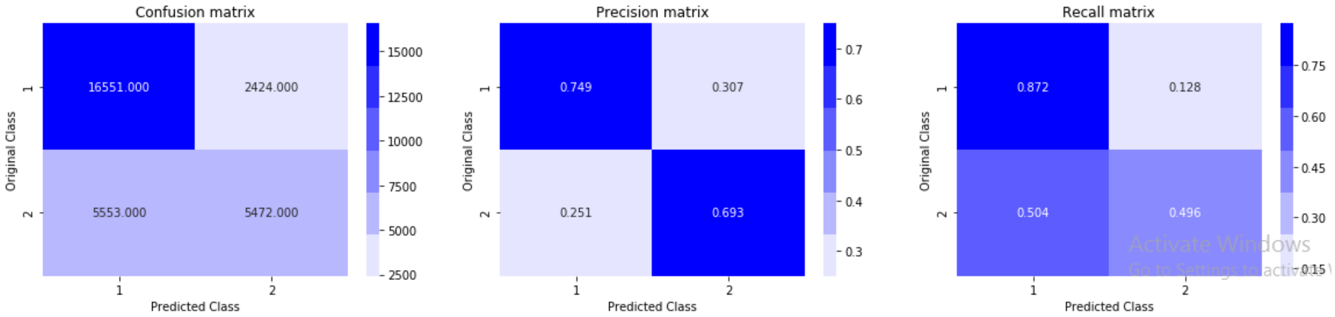

**Fig 4.** Confusion Matrix, Precision Matrix and Recall Matrix for Logistic Regression

## 3.3 Linear SVM

Figure 4 exhibits the confusion matrix, precision matrix and recall matrix for the linear SVM model. The confusion matrix confirms that true positive has the highest value at 17081. The log loss (error function) obtained for the linear SVM model is 0.654 which is smaller than the error acquired for the random forest but greater than the error obtained for logistic regression. Thus, this model is better in comparison to the random forest while it is not as good as the Logistic Regression. Figure 5 analyses the Cross-Validation Error for each alpha taken. It is an alpha versus error measure graph. It is used to find the optimum value of alpha.

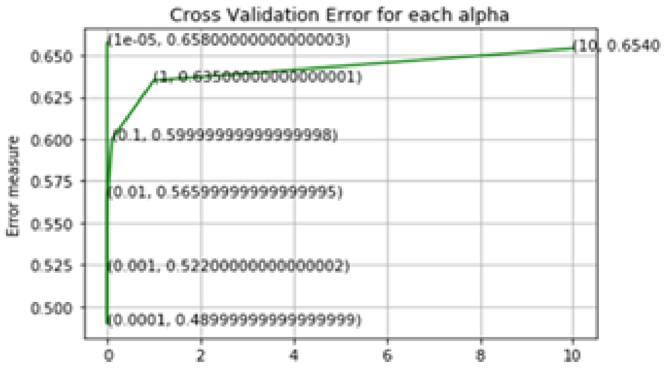

**Fig 5.** Cross-Validation for Linear SVM

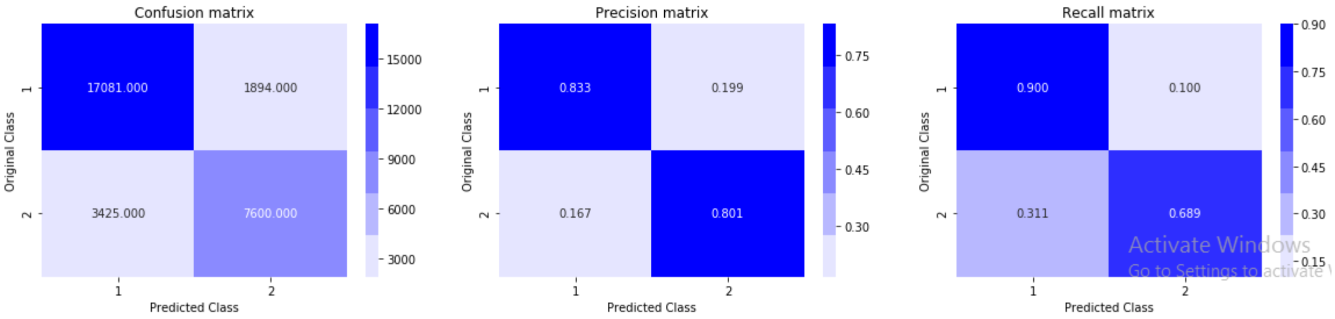

**Fig 6.** Confusion Matrix, Precision Matrix and Recall Matrix for Linear SVM

### 3.4 XGBoost

Figure 7 explains values acquired for confusion matrix, precision matrix and recall matrix for XGBoost. With regard to the confusion matrix, it is definite that the value obtained for true positive (i.e. the actual value is yes and the predicted value is also yes) is at the peak which corresponds to 17172. It is evident that the true positive value is the highest for XGBoost when compared to all the previous models. The log loss (error function) for XGBoost is found to be 0.357 which is the least among the four models. Since error function is inversely proportional to efficiency, it is clear that the results obtained by using the XGBoost approach is the most accurate when compared to the above three models.

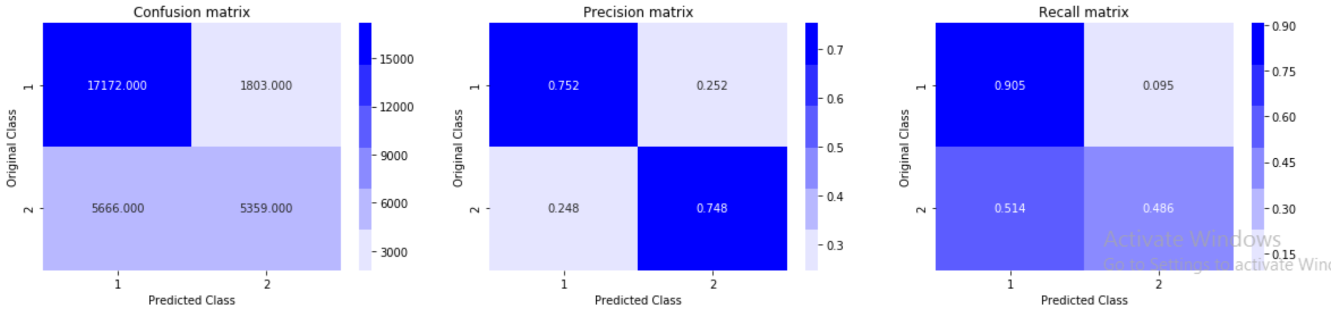

**Fig 7.** Confusion Matrix, Precision Matrix and Recall Matrix for XGBoost

## 4 Conclusion

This study trained and tested four machine learning models to identify duplicate questions using a real-time dataset. The size of the raw dataset was found to be 7GB. PL/SQL was used to pre-process data which was further stored in the database. PL/SQL loads the entire dataset only once and whenever any query is executed, data is directly obtained from the database, therefore this process is rapid and efficient. The entire dataset was successfully cleaned and pre-processed in one hour. While the existing solutions use python functions available in the python libraries to pre-process huge datasets which take four times the time taken by PL/SQL.

Four different machine learning models were implemented and their results were compared to arrive at the best-performing model. After execution, the error parameter referred from the log loss function for the random model, logistic regression model, linear SVM and XGBoost are obtained to be 0.887, 0.521, 0.654 and 0.357. Efficiency is inversely proportional to the error function thus it can be concluded that XGBoost is the best model, yielding maximum accuracy in the minimal amount of time which is complemented by the unique pre-processing operations performed using PL/SQL thereby improving the overall response time.

## Future Scope

As a continuation to our project, in the future, a similar approach can be implemented for various other search engines that are available like reedit, stack overflow, etc. This will ensure that search engines are user-friendly. Furthermore, the best models of each type (linear, tree-based, and neural network) can be grouped. Additionally, more features for the tree-based models can also be designed.

## References

1) Imtiaz Z, Umer M, Ahmad M, Ullah S, Choi GS, Mehmood A. Duplicate Questions Pair Detection Using Siamese MaLSTM. *IEEE Access*. 2020;8:21932–21942. Available from: https://dx.doi.org/10.1109/access.2020.2969041.
2) Li W, Peng X, Cheng K, Wang H, Xu Q, Wang B. A Short-Term Regional Wind Power Prediction Method Based on XGBoost and Multi-stage Features Selection. *2020 IEEE 3rd Student Conference on Electrical Machines and Systems (SCEMS)*. 2020;p. 614–618. Available from: 10.1109/SCEMS48876.2020.9352249.
3) Chen M, Liu Q, Chen S, Liu Y, Zhang C, Liu R. XGBoost-Based Algorithm Interpretation and Application on Post-Fault Transient Stability Status Prediction of Power System. *IEEE Access*. 2019;7:13149–13158. Available from: 10.1109/ACCESS.2019.2893448.
4) Sultana R, Rawat S, Murthy GV, Kumar N. An Investigation on Managing Patient Flow at Hospital Emergency Care Unit Using Tree-Based Data Mining Techniques. In: R C, S D, S R, editors. Advances in Computational Intelligence and Informatics. ICACII 2019;vol. 119. Springer. 2020;p. 237–243. Available from: https://doi.org/10.1007/978-981-15-3338-9_28.
5) Cherif IL, Kortebi A. On using eXtreme Gradient Boosting (XGBoost) Machine Learning algorithm for Home Network Traffic Classification. In: and others, editor. 2019 Wireless Days (WD). 2019;p. 1–6. Available from: 10.1109/WD.2019.8734193.
6) Dong X, Lei T, Jin S, Hou Z. Short-Term Traffic Flow Prediction Based on XGBoost. *2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS)*. 2018;p. 854–859. Available from: 10.1109/DDCLS.2018.8516114.
7) Saedi C, Rodrigues J, Silva J, Branco A, Maraev V. Learning Profiles in Duplicate Question Detection. *2017 IEEE International Conference on Information Reuse and Integration (IRI)*. 2017;p. 544–550. Available from: 10.1109/IRI.2017.39.
8) Xu Z, Yuan H. Forum Duplicate Question Detection by Domain Adaptive Semantic Matching. *IEEE Access*. 2020;8:56029–56038. Available from: 10.1109/ACCESS.2020.2982268.
9) Wang L, Zhang L, Jiang J. Duplicate Question Detection With Deep Learning in Stack Overflow. *IEEE Access*. 2020;8:25964–25975. Available from: 10.1109/ACCESS.2020.2968391.
10) Prabowo DA, Budi G, Herwanto. Duplicate Question Detection in Question Answer Website using Convolutional Neural Network. *2019 5th International Conference on Science and Technology*. 2019;p. 1–6. Available from: 10.1109/ICST47872.2019.9166343.
11) Mukherjee S, Kumar NS. Duplicate Question Management and Answer Verification System. *2019 IEEE Tenth International Conference on Technology for Education (T4E)*. 2019;p. 266–267. Available from: 10.1109/T4E.2019.00067.
12) Daoud M. Novel Approach towards Arabic Question Similarity Detection. *2019 2nd International Conference on new Trends in Computing Sciences (ICTCS)*. 2019;p. 1–6. Available from: 10.1109/ICTCS.2019.8923102.
13) Ye B, Feng G, Cui A, Li M. Learning Question Similarity with Recurrent Neural Networks. *2017 IEEE International Conference on Big Knowledge (ICBK*. 2017;p. 111–118. Available from: 10.1109/ICBK.2017.46.
14) Shankar S. Identifying Quora question pairs having the same intent. 2017.
15) Mahmood Q, Qadir MA, Afzal MT. Document similarity detection using semantic social network analysis on RDF citation graph. *2013 IEEE 9th International Conference on Emerging Technologies (ICET)*. 2013;p. 1–6. Available from: 10.1109/ICET.2013.6743548.
16) Wang J, Li Z, Hu B. A context approach to measuring similarity between questions in the community-based QA services. *Seventh International Conference on Fuzzy Systems and Knowledge Discovery*. 2010;p. 2408–2411. Available from: 10.1109/FSKD.2010.5569521.