

Development of Answer Validation System Using Responders' Attributes and Crowd Ranking

Mercy Adebisi¹, Bolanle Ojokoh^{2*} , Tolulope Adebayo¹, Akintoba Akinwonmi¹, Fatai Sunmola³

¹Department of Computer Science, Federal University of Technology, Akure, Nigeria

²Department of Information Systems, Federal University of Technology, Akure, Nigeria

³Department of Information Technology, Federal University of Technology, Akure, Nigeria

Email: adebisimercy4real@gmail.com, *bolanleojokoh@yahoo.com, toluadebayo4@gmail.com, aeakinwonmi@futa.edu.ng, fosunmola@futa.edu.ng

How to cite this paper: Adebisi, M., Ojokoh, B., Adebayo, T., Akinwonmi, A., & Sunmola, F. (2021). Development of Answer Validation System Using Responders' Attributes and Crowd Ranking. *Journal of Service Science and Management*, 14, 382-398.

<https://doi.org/10.4236/jssm.2021.143024>

Received: February 22, 2021

Accepted: June 27, 2021

Published: June 30, 2021

Copyright © 2021 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Crowdsourcing has found a wide range of application in Community Question Answering (CQA). However, one of its biggest challenges is the need to address the quality of crowd answers contributions. Therefore, this work proposed a system that seeks to validate answers to questions provided by respondents using responders' attributes and crowd ranking technique. Weights were assigned to respondent answers based on their academic records, experience and understanding of the question to obtain valid answers. Thereafter, valid answers were ranked by the crowd using Borda Count algorithm. The proposed system was evaluated using Usability and User experience (UX) measurement. The result obtained demonstrated the effectiveness of the applied technique.

Keywords

Askers, Answerers, Community Question Answering (CQA), Question Answering (QA), Crowd Sourcing, Answer Validation

1. Introduction

The new information era provides readily available access to information, especially with the advent of the internet. Different questions requiring correct answers are uploaded on the internet on daily basis which leads to the development of question answering (QA) systems, with the aim of providing accurate answers to explicit questions which are contrasting to document retrieval (Ojokoh &

Adebisi, 2019; Toba et al., 2014). Schofield and Thielscher (2019) defined community QA as a website or service that requires a method to display pieces of information in the form of a question in natural language, a medium for communal response and a community in which questions and answers are rooted based on the level of participation, and answers provided was discovered to be of higher quality when it was compared with other types of online QA services (Harper et al., 2008). However, answers to questions from users form the pillar of a successful CQA service, in which better answers may be provided as against automatic systems. However, while the attitude and the reliability of users on the web vary, the quality of answers provided may not be of high quality, and this prompted the idea of answer validation by accessing the correctness of answers provided by respondents using different techniques (Magnini et al., 2002, 2005). Validation of answers became essential because crowdsourcing tasks providers have restricted control over the selection of crowd workers and little insight into the level of know-how and dependability of the workers that provide answers. Crowdsourcing as defined by Howe (2006) is an act of farming out a job ordinarily performed by a selected employee to an open-ended large group of people usually in the form of an open call. The performance of these crowd workers largely determines the worth of the result obtained from a task. Hung et al. (2015) described five types of crowd workers as: *Reliable workers* (having profound knowledge about specific fields and give response to questions with very high reliability, in that all the answers given by them are correct). *Normal workers* (have general knowledge to respond to questions, but seldom make mistakes, that is, three out of four of their answers are correct). *Sloppy workers* (have very miniature knowledge thereby providing erroneous answers, however unintentionally). *Uniform spammers* (who intentionally give the same answer for all questions). *Random spammers* (who imprecisely give casual and worthless answers for all).

Several studies have been carried out on how to make better the quality of the answers provided by QA system, focusing on textual entailment, question type analysis, answer ranking by the crowd workers and domain experts and personal and community features (past history) of the answerer to determine the quality of the answers (Ríos-Gaona et al., 2012; Su et al., 2007; Ishikawa et al., 2011; Ojokoh & Ayokunle, 2012; Anderson et al., 2012; Schofield & Thielscher, 2019). Since past history alone may not be fitting enough to determine the quality of an answer, level of confidence in the answer provided is introduced in order to obtain credible answers from respondents. The proposed system is aimed at using community presence interaction as one of the basis for quality answer selection; capturing crowd specialty as part of the personal features used to validate answers; modelling the criteria used in evaluation automatically and preventing bias crowd ranking of answers by enabling them to specify their preferential schedule using Naïve Bayes Spam filter and Borda count ranking Algorithm.

The remaining part of this paper is structured as follows: Section 2 presents

the review of related works. Section 3 presents the proposed system architecture, and the description of the components that make up the architecture. Section 4 is dedicated to the experimental setup and results while Section 5, concludes the paper and presents some future works.

2. Related Works

Question Answering (QA) according to [Chandra et al. \(2017\)](#) is a computer science discipline concerned with developing a system that automatically provide answers to questions requested by human in a natural language. QA study attempts to deal with a wide-ranging question types that consist of facts, lists, definitions, how, why, putative, semantically constrained, and cross lingual questions ([Cimiano et al., 2014](#)). [Ishikawa et al. \(2011\)](#) manually chose questions and answers at random from Yahoo archives, which were evaluated by four assessors to identify evaluation criteria. These criteria were later used to construct a model to identify high-quality answers. [Šimko et al. \(2013\)](#) presented a method for validating question-answer learning objects involving interactive exercise for learners by employing students' accuracy estimations of answers provided by other students, during learning. The method was deployed within an adaptive Learning Framework and they were able to show that total student crowd estimations are to a great extent analogous to teacher's assessment of provided answers. [Aydin et al. \(2014\)](#) presented a method to integrate crowdsourcing and Machine Learning (ML) techniques in order to develop a crowd-sourced "Who Wants to Be a Millionaire" player quiz show. They employed lightweight machine learning techniques to improve the combined correctness of crowd-sourced answers to questions. The results showed improvement in the success rates of the harder questions by investigating new weighted aggregation patterns for answers obtained from the crowd and they were able to build a super player for the game that can provide answers to questions from all difficulty levels with a precision of above ninety percent (90%).

[Dobšovič et al. \(2014\)](#) proposed and developed a CQA system "Askalot" which is focused on the area of education by implementing a functionality that encompasses the educational goal and specifics of universities, based on open source technologies. Answers to questions are verified by other students, comments are however provided by a teacher using a five-grade scale on which the assessment of the quality of question or answer can be done. [Toba et al. \(2014\)](#) proposed a hybrid hierarchy-of-classifiers framework to model QA pairs and integrate the question type analysis and answer quality information in an integrated framework. The quality classifier gives two probabilities each, showing the probability of good or bad-quality. They tested the framework on a dataset of about 50 thousand QA pairs from Yahoo! Answers and an effective identification of high quality answers was realized based on their evaluation of the system. [Tran et al. \(2015\)](#) presented a method to detect the right or possible right answers from the answer thread in Community Question Answering pools. They

used multiple features for quality answer selection which exploits the surface word-based similarity between the question and answer to allot score using a regression model. Afterwards, translation probabilities were computed via IBM and Hidden Markov Models to obtain the likelihood of an answer being the translation of the question. [Savenkov et al. \(2016\)](#) presented a system that could be used to filter or re-rank the candidate answers by providing validation for the answers. They specifically focused on knowing the effect of time restrictions in the close real-time QA setting, thereby developing a way in which crowd will be able to create the answer candidates directly within a limited amount of time and also the way in which crowd will be able to rank sets of given answers to a question within a specified amount of time. [Hung et al. \(2017\)](#) developed a probabilistic model that helps to recognise the most valuable validation questions in improving results' accuracy and detecting faulty workers in their quest to validate and control the quality of crowd answers to reduce cost incurred from utilizing experts.

[Nie et al. \(2017\)](#) presented a novel scheme to rank answer candidates via pairwise comparisons consisting of one offline learning and one online search component. In the online search component, a pool of candidate answers for the given question was extracted via finding its similar questions. The extracted answers were then sorted by leveraging the offline trained model to judge the preference orders.

[Fan et al. \(2019\)](#) proposed to enhance answer selection in CQA using multidimensional feature combination and similarity order. They made full use of the information in answers to questions to determine the similarity between questions and answers, and use the text-based description of the answer to determine its sensibility. [Le et al. \(2019\)](#) proposed a framework for automatically assessing answer quality by integrating different groups of features such as personal, community-based, textual, and contextual, to build a classification model and determine what constitutes answer quality. Experiments conducted on Brainly and stack overflow datasets show that the random forest model achieves high accuracy in identifying high-quality answers. Also indicating that personal and community-based features have more prediction power in assessing answer quality.

In this paper, we leverage on the fact that the performance of the crowd workers determines the quality of the result of a crowdsourcing task, and hence the need to develop an effective and reliable question answering system that is capable of validating and evaluating the answers provided by the crowd because of their varying reliability as established in past works ([Hung et al., 2017](#); [Savenkov et al., 2016](#)). All these are important issues to be addressed in Artificial Intelligence.

3. The Proposed System

The architectural overview of the proposed system is presented in [Figure 1](#). The subsections that follow describes each of the segments.

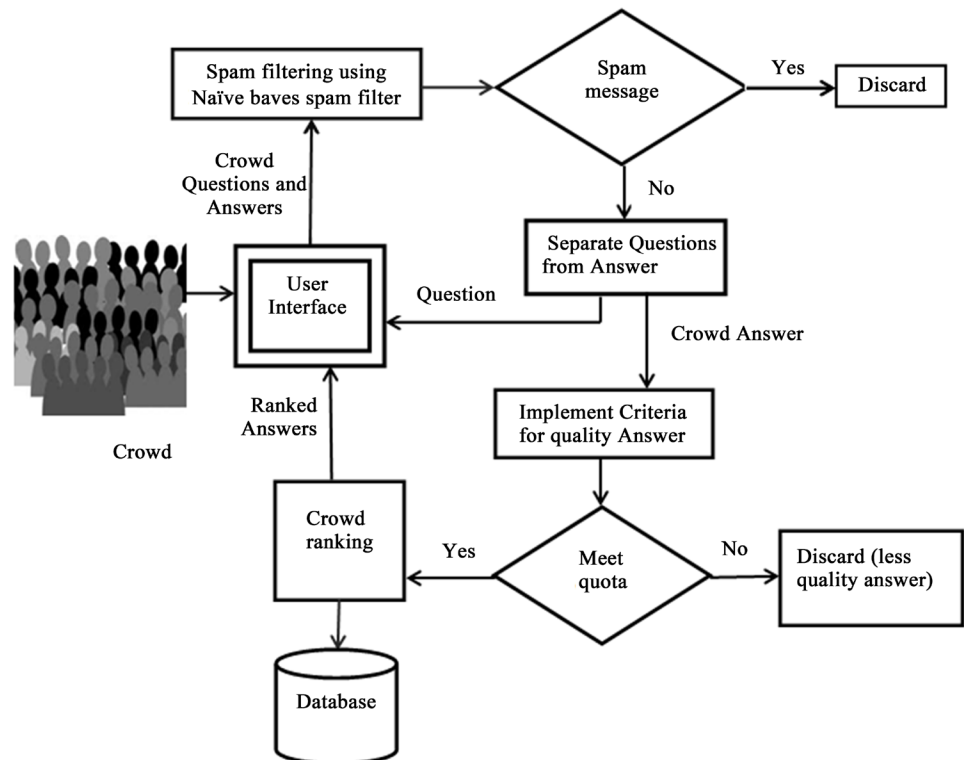


Figure 1. Proposed system architecture.

3.1. User Interface

The user interface module consists of four (4) components listed as follows:

- 1) **Ask Question:** This component enables the asker (that is someone who wishes to ask any computer-related questions) to post his/her questions on the platform.
- 2) **Answer Question:** This component enables experts or anyone familiar with the question asked to provide answers.
- 3) **Rank Answers:** This allow users from the crowd to rank answers provided by other users based on their knowledge of the question.
- 4) **View Recent Questions:** This component provides a view of the list of the most recently posted questions.

3.2. Database

The database is the component of the Answer Validation model that stores information about the system and its users. It stores both legitimate questions and answers from web users, and most importantly, answerers' personal information for the purpose of validating their answers which is obtained the first time a respondent uses the system.

3.3. Naïve Bayes Spam Filter

Naïve Bayes (NB) Spam Filter, a machine learning algorithm, which is one of the powerful tools for Artificial Intelligence was used in this work to filter inconse-

quential and redundant messages from the collection of messages or information provided by the crowd. Every incoming text (both question and answer) pass through the trained Naïve Bayes Spam filter to determine the probability of the message being a legitimate message or spam. The NB spam filter is trained with the commonly used online spam words and spam dataset downloaded from kaggle.com. A sample is shown in **Figure 2**.

From Bayes' theorem, the probability that a message with vector $X = (X_1, \dots, X_m)$ belongs in category c is:

$$P(c|x) = \frac{p(c) \cdot p(x|c)}{p(x)} \quad (1)$$

Using Naïve Bayes Spam filter, a message is classified as spam whenever

$$P = \frac{p(c_s) \cdot p(x|c_s)}{p(c_s) \cdot p(x|c_s) + p(c_h) \cdot p(x|c_h)} \quad (2)$$

$$P \begin{cases} > T, \text{ message is spam} \\ \leq T, \text{ message not spam} \end{cases} \quad (3)$$

where c_s is a message in spam category; c_h is a message in ham category;

$p(c_s)$ is the probability that the response x belongs to spam category, c_s ;

$p(c_h)$ is the probability that the response x belongs to ham category, c_h ;

$p(x|c_s)$ is the likelihood of response x given the spam category;

$p(x|c_h)$ is the likelihood of response x given the ham category and;

T is a threshold value.

If P is greater than T , the incoming message is being classified as spam message and will be discarded else if P is less than or equal to T , the message will be accepted by the system and presented as a question or accepted as an incoming answer.

3.4. Separate Question from Answer

This is the component of the system where a legitimate message from the user is being identified as either a question or answer. If the incoming message is a


	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Label	Message												
2	spam	Dear Customer, You have a missed call from +917985862318 . The last missed call was at 11:18 PM on 14-Oct-2017 . Thankyou, Team Jio.												
3	 (Ctrl) ▾	Join V-STUDY and score excellent marks in class 12th (Our Students feedback-95%+ coverage from our assignments,class room example												
4	spam	Join crash courses for B.ST,A/C'S,ECO,ENG,&IP (fee only Rs.5000 per sub) from expd faculties.Branches in Sec-3 Rohini & Pitampura. Call												
5	spam	CRASH COURSES by BEST POOL OF FACULTY. ENGLISH by MA(ENG) MEGHA SURI, ACCOUNTS by CS NITIN GUPTA,ECO by CS NITIN, IP by M												
6	spam	Dear Ola Shuttle user, get 60% Off on your next 2 shuttle rides. Use code: SHUTTLE60. Valid for first 5,000 users. Book Now !												
7	spam	Please pay bill amount of Rs. 29 in cash for your Ola ride, served by Rajendra.												
8	spam	Watch the undercover story of Subhash Chandra Bose. Unveiling Indias biggest mystery BOSE: DEAD OR ALIVE watch now on Vodafone P												
9	spam	Bollywood Beauties go Bold. Get Videos, Wallpapers of Bold Bollywood Beauties. Click http://hng.am/8291												
10	spam	Ab hum laye hai aapke liye romantic calletunes ka bhandaar,Download kare Callertune app Click http://hyperurl.co/ctt aur payein FREE												
11	spam	Govt Mandate: Update Aadhaar to avoid blocking of your Axis Bank Account by 31/03/18. Visit your branch or click goo.gl/d214uS . Kindly												
12	spam	You have used 50% of your 1 GB daily high speed internet quota on your Jio Number 8178390589 as of 18-Nov-17 20:57 Hrs. To track your												
13	spam	You have used 50% of your 1 GB daily high speed internet quota on your Jio Number 8178390589 as of 04-Dec-17 13:57 Hrs. To track your												
14	spam	You have used 50% of your 1 GB daily high speed internet quota on your Jio Number 8178390589 as of 07-Jan-18 20:52 Hrs. To track your												

Figure 2. Spam dataset from <https://www.kaggle.com/>.

question, this component ensures that the question is presented at the User Interface for the answerers to provide answers, and if otherwise, the system will pass it to the next component where the criteria for quality answers will be implemented.

3.5. Criteria for Quality Answers

The quality of the result of a question answering system rest on the source of the answers provided by the system. Since the aim of the question answering system is to provide a precise answer in natural language; it is therefore important to provide quality assurance on every answer obtained from the web users, as these users can vary in reliability. The criteria employed for validation and used to ensure quality answers in this work are User attributes, Area of Specialization, Understandability and Confidence (displayed in **Table 1**).

3.6. Weighted Voting System

A game playing situation is applied for ranking answers using a collection of weighted players P_i together with a quota q , which is the total number of votes required to pass a motion. This is used to determine the level of reliability of the users that provide answers. A player is a user attribute that is used to allot point to answerers. In a weighted voting system, a player's weight w_i refers to the number of points allotted to that player and is always a positive integer value. A weighted voting system is described by specifying the voting weights, w_1, w_2, \dots, w_n of the players P_1, P_2, \dots, P_n , and the quota, q . A coalition is called winning if the sum of the players' weights is greater or equal to the quota, and losing if otherwise. The coalitions, which are the criteria used in this work to ensure quality answers from the web users are User attributes, area of specialization, Understandability and Confidence. User attributes that are used comprises of user Course of study, Grade point, number of years of experience in computing and the general level of knowledge of computing. Point is added to the weight of the responder based on their selections from the range of value of the attributes. A user is also allowed to choose any area of specialization such as Networking, Cyber Security and hardware and repairs and so on. Users' understandability of the given question is measured based on a five-level rating scale, as well as the Confidence which is a way in which the answerer can infer how much the system can trust the answer provided. This is also measured based on a five level rating scale. Combining these and the weighted voting system, this phase of the system is represented by:

$$q : P_1, P_2, P_3, P_4$$

where,

P_1 is User's personal attribute, P_2 is Specialization;

P_3 is Understandability, P_4 is Confidence.

The totality of weights, T_w per Answerer is computed as:

Table 1. Weight distribution table.

S/N	Criteria for quality Assurance	Description	Metrics	Points
1	User attributes (P_1).	Grade point.	First class	5
			Second class upper	4
			Second class lower	3
			Third class	2
			Pass	1
		Course of study during Undergraduate.	Computer Science related course	5
			Other science related course	3
		Numbers of Years of experience.	21 yrs and above	5
			16 - 20 yrs	4
			11 - 15 yrs	3
			6 - 10 yrs	2
			1 - 5 yrs	1
		General level of computing.	Very high	5
			High	4
			Medium	3
			Low	2
			Very low	1
2	Area of specialization (P_2)	Area of specialization of answerers.	Specialize area	5
			Non-specialize area	3
3	Understandability (P_3)	Level of Understanding of the question by the answerers	Very high	5
			High	4
			Medium	3
			Low	2
			Very low	1
4	Confidentiality (P_4).	Level of Confidence of the answerers in their answer	Very high	5
			High	4
			Medium	3
			Low	2
			Very low	1

$$T_w = \sum_{i=1}^4 w_i \quad (4)$$

where w_i is the weight corresponding to each player, P_i . The maximum weight, N obtainable by an answerer with q being the minimum weight required for an acceptable (valid) answer is expressed as:

$$N = w_1 + w_2 + w_3 + w_4 \quad (5)$$

$$\text{then, } \frac{N}{2} < q \leq N \text{ holds for equation} \quad (6)$$

In this work, q was obtained by calculating the 70% of N as follows:

$$q = 70\% N$$

From Equation (6), q can be said to be less than or equal to N but greater than $\frac{N}{2}$. This means that $\frac{35}{2} < q \leq 35$. Since this work is based on quality answer validation, 70% of N was used as the quota q .

$$\text{Quota}(q) = \frac{35}{100} \times 70 = 24.5 = 25 (\text{approx.})$$

Therefore the quota, q will be 25. **Table 2** depicts the different criteria considered in this work with the respective maximum weight obtainable.

Depending on the point obtained from each criterion by the Responders (Answerers), these points are aggregated based on their selection. The total weight of the answer is calculated to check whether the weight meets up to the quota. If the total weight of the answer is greater or equal to the quota, the answer is considered valid and is passed to the next phase which is the ranking phase, and if not the answer is discarded.

3.7. Crowd Ranking

The last phase employs a crowdsourcing ranking algorithm called Borda count. The algorithm ranks all the valid answers from phase two using a preference schedule point. It awards points to candidates based on preference schedule, then the candidate with the highest points is declared the winner. For instance, given M , the number of candidate answers, each first-place, second-place and third-place votes is worth $M, M-1, M-2$ points respectively. Consequently, each M th-place (that is, last-place) vote is worth 1 point. Now, suppose there are n voters, every voter ranks the M candidates according to his preference, and a candidate answer has an average rank score, s_n .

$$s_n = \sum_{i=1}^n r_i \quad (7)$$

where r_i is the point assigned by n crowd (ranker).

Table 2. Maximum weight obtainable (N).

S/N	Criteria	Maximum weight Obtainable (N)
1.	User Attributes (P_1)	20
2.	Area of specialization (P_2)	5
3.	Users understandability (P_3)	5
4.	Users confidentiality (P_4)	5
Total		35

The candidate answers will be ranked according to their performance starting from the best on top of the list (answer with the highest point) to the worst (answer with the lowest point).

4. Experiments and Evaluation

4.1. Data and Tools

A dataset consisting of 185 Spam messages was downloaded from Kaggle.com and was used to train the Naïve Bayes Filter in order to distinguish between legitimate and inconsequential information provided by the crowd. The system was implemented using HTML, Python Script and Django web framework.

4.2. Experimental Setup

Experiments were conducted to verify the system performance and to determine how useful and precise the answers provided were. The users of the system are allowed to post questions which will be answered by responders who are vast in the field of the question being asked. However, before the responders would be allowed to provide answers, they will be required to sign/sign up as the case may be, verifying their Course of study, Area of specialization, Grade point, number of years of experience in Computing, general level of Computing knowledge and the level of understanding of the question. Also, the confidence level of the responder will be confirmed before posting the answer. In cases where a minimum of five different answers are provided to a particular question, they are ranked by the crowd starting from the most correct to the least correct answer. A sample of asked questions and answers provided is shown in **Figure 3**.

QUESTION	ANSWER	RANKS	POINTS	BY	DATE
How good of a programmer is Mark Zuckerberg and does he still sometimes code for Facebook?	He is one of the best programmer when you are talking about explicit coding. im sure of that.	0	32	adebisi asiat mercy	Dec. 20, 2018, 4:01 a.m.
How can i learn a web design in a day?	Web design learning can start and end whenever you feel like stop learning. A day may not be enough to learn all about web design. But a website can be designed in day and be expanded later.	0	29	Thomas	Oct. 28, 2018, 4:25 a.m.
How good of a programmer is Mark Zuckerberg and does he still sometimes code for Facebook?	He is a good programmer and still very much involve in every development of Facebook. Though not sure if he still code for Facebook.	0	29	Thomas	Oct. 28, 2018, 4:16 a.m.
Do mac address always stay the same?	Yes. They are inscribed by the manufacturer so it is static.	0	32	Josephine	Oct. 25, 2018, 9:34 p.m.
How can i learn a web design in a day?	Web design is a very technical concept and I believe it can't all be leant in a day but the process of learning it can begin in a day	0	29	Adewole Victor	Oct. 24, 2018, 3:22 p.m.
How good of a programmer is Mark Zuckerberg and does he still sometimes code for Facebook?	He's very good and I believe he still codes for facebook	0	28	Adewole Victor	Oct. 24, 2018, 3:20 p.m.

Figure 3. Sample of questions and answers.

4.3. Evaluation

The method of evaluation used in this work is based on ISO/IEC 9126 standard metrics and the Usability and User experience (UX) measurement instruments adopted in (Tan et al., 2010). The model consists of 21 subcharacteristics distributed on six main characteristics of software measurement metrics. Using the common Goal Question Metric (GQM) approach, a nomenclature for usability and UX attributes were defined and were able to identify an extensive set of questions and measures for each attribute. The metrics used for this work are shown in **Table 3**.

From the above stated metrics, twenty (20) questions were formed in order to evaluate the Answer Validation system by Users. Eighty five users out of One hundred sample size evaluated the system, with each question (Q_1, \dots, Q_{20}) answered using four-level rating scale; Very High, High, Medium and Low respectively. Ratings obtained from the Users were analyzed using weight means techniques in which weights are added (such that Very High = 4, High = 3, Medium = 2 and Low = 1) to users feedback. A sample of the questionnaire is shown in **Table 4**.

4.4. Results and Discussion

The ratings were analyzed and the frequency at which each point occurs was obtained. The metrics were measured and analyzed to form a continuous score in percentage (%). **Table 5** illustrates the number of users out of eighty-five (85) that rated the system either Very high, High, Medium or Low based on the given questionnaire. **Figure 4** and **Figure 5** shows the graphical representation of the obtained results. **Table 6** shows the Combination of Very High and High ratings in order to define the User ratings as High, Medium, Low. **Figure 6** and **Figure 7** show the Combination of Very High and High ratings for Usability and User Experience respectively.

Table 3. Usability and user experience metrics.

S/N	Usability metrics	User Experience (UX) metrics
1.	Understandability	Correctness
2.	Efficiency	Satisfaction
3.	Error tolerance	Simplicity
4.	Ease of use	Validation
5.	Attractiveness	Effectiveness
6.	Time response	Quality of outcome
7.	Visualization	Reliability
8.	Navigability	Consistency
9.	Reusability	Accessibility
10.	Feedback	Preferability

Table 4. Questionnaire for answer validation system evaluation.

S/N Usability metrics		User Experience (UX) metrics
1.	What is the rate at which you understand the system?	What is the rate at which the answers provided by the system are correct?
2.	What is the rate at which you think the system is efficient?	What is the rate at which you are satisfied with the answers provided by the system
3.	What is the rate at which the system tolerates error and corrects you when you made mistakes?	What is the rate at which the language used by the system is simple?
4.	What is the rate at which the system is easy to use?	What is the rate at which the answers provided by the system in corresponding to their questions are valid?
5.	What is the rate at which the system design is attractive?	What is the rate at which the system is effective enough in providing valid answer to questions?
6.	What is the rate at which you are satisfied with the time response of the system?	What is the rate at which the system can provide high quality answers?
7.	What is the rate at which you are satisfied with the visual content of the system?	What is the rate at which the system is reliable in providing answers to computing related questions?
8.	What is the rate at which you find it easy to Navigate through the system?	What is the rate at which the system is consistent in performing its functions?
9.	What is the rate at which you will like to use the system the next time?	What is the rate at which the system is accessible from your end?
10.	What is the rate at which you are satisfied with the system feedback?	What is the rate at which you prefer the system to others?

Table 5. User rating frequency table and their percentage.

Metrics/Ratings	Very High		High		Medium		Low	
Correctness	44	51.76%	38	44.71%	3	3.53%	0	0%
Satisfaction	46	54.12%	39	45.88%	0	0%	0	0%
Validation	43	50.59%	40	47.05%	2	2.35%	0	0%
Effectiveness	30	35.29%	50	58.82%	5	5.88%	0	0%
Quality of Outcome	35	41.17%	48	56.47%	2	2.35%	0	0%
Reliability	24	28.24%	60	70.58%	1	1.18%	0	0%
Consistency	58	68.24%	25	29.41%	2	2.35%	0	0%
Accessibility	42	49.41%	42	49.41%	1	1.18%	0	0%
Fault Tolerance	20	23.53%	57	67.05%	8	9.41%	0	0%
Preferability	18	21.17%	65	76.47%	2	2.35%	0	0%
Ease of use	34	40.0%	50	58.82%	1	1.18%	0	0%
Navigability	48	56.47%	34	40.00%	3	3.53%	0	0%
Simplicity	30	35.29%	53	62.35%	2	2.35%	0	0%
Understandability	39	45.88%	45	52.94%	3	3.53%	0	0%
Attractiveness	38	44.71%	39	45.88%	8	9.41%	0	0%
Time Response	32	37.65%	28	32.94%	25	29.41%	0	0%
Visualization	20	23.53%	56	65.88%	9	10.58%	0	0%
Reusability	20	23.53%	62	72.94%	3	3.53%	0	0%
Feedback	21	24.71%	54	63.53%	10	11.76%	0	0%
Efficiency	38	44.71%	44	51.76%	3	3.53%	0	0%
Total	680		929		93		0	

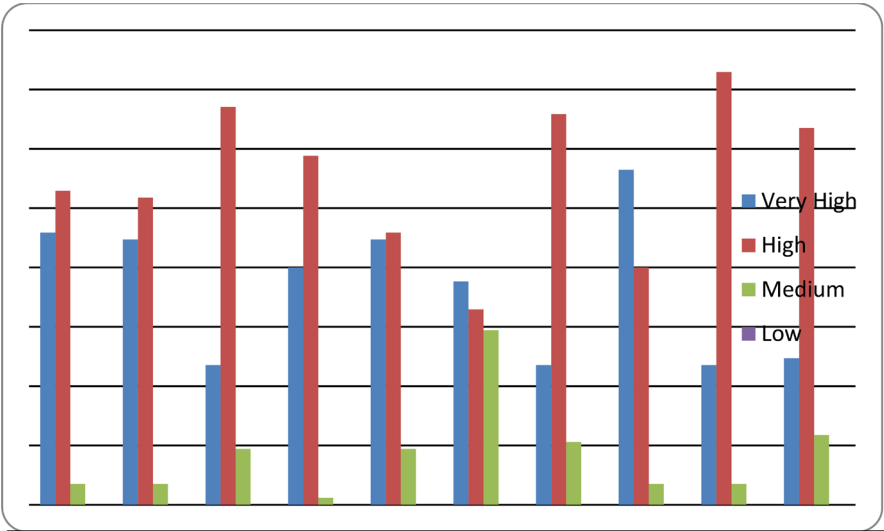


Figure 4. Usability graph.

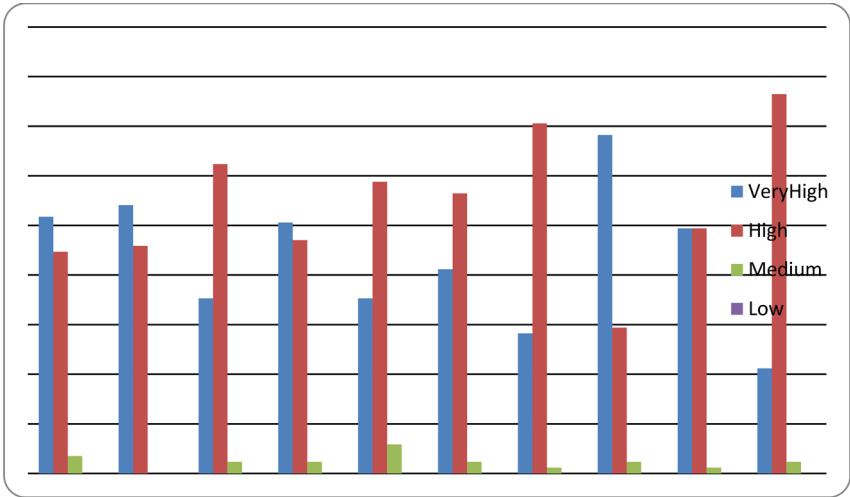


Figure 5. User experience graph.

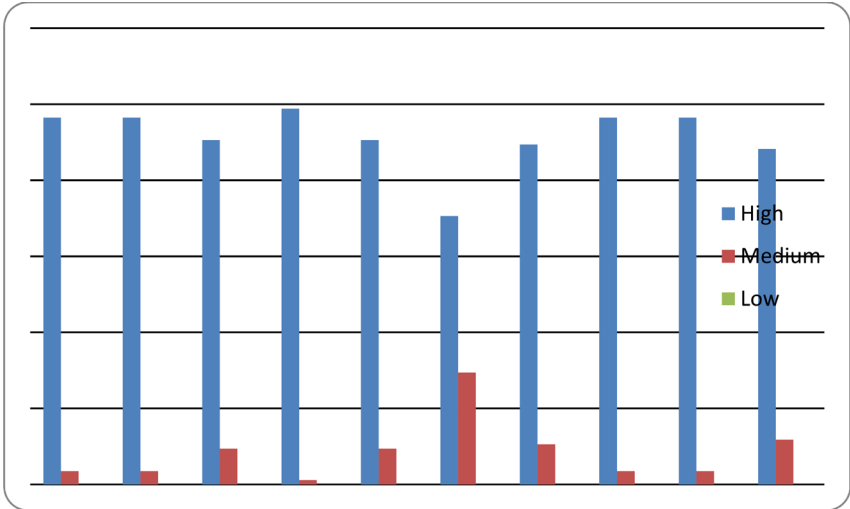


Figure 6. Combined very high and high rating for usability.

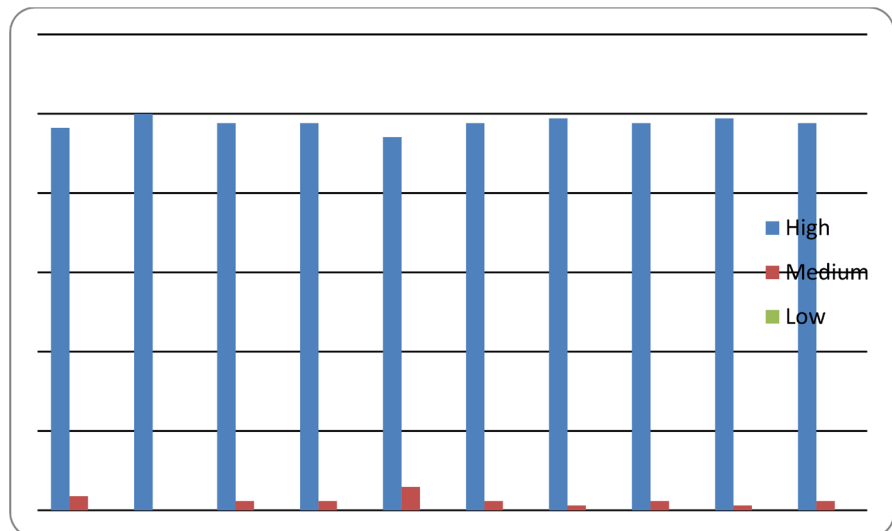


Figure 7. Combined very high and high rating for user experience.

Table 6. Combined very high and high rating.

Metrics/Ratings	Combined High		Medium		Low	
Correctness	82	96.47%	3	3.53%	0	0%
Satisfaction	85	100%	0	0%	0	0%
Validation	83	97.65%	2	2.35%	0	0%
Effectiveness	80	94.11%	5	5.88%	0	0%
Quality of Outcome	83	97.65%	2	2.35%	0	0%
Reliability	84	98.82%	1	1.18%	0	0%
Consistency	83	97.65%	2	2.35%	0	0%
Accessibility	84	98.82%	1	1.18%	0	0%
Fault Tolerance	77	90.59%	8	9.41%	0	0%
Preferability	83	97.65%	2	2.35%	0	0%
Ease of use	84	98.82%	1	3.33%	0	0%
Navigability	82	96.47%	3	3.53%	0	0%
Simplicity	83	97.65%	2	2.35%	0	0%
Understandability	82	96.47%	3	3.53%	0	0%
Attractiveness	77	90.59%	8	9.41%	0	0%
Time Response	60	70.59%	25	29.41%	0	0%
Visualization	76	89.41%	9	10.58%	0	0%
Reusability	82	96.47%	3	3.53%	0	0%
Feedback	75	88.23%	10	11.76%	0	0%
Efficiency	82	96.47%	3	3.53%	0	0%
Total	1607		93		0	

The overall results show that the user experience evaluations of the system based on the metrics given are excellent. This is because in most case of the metrics used “Very High” and “High” (which are good scale to measure superior or improved opinion) are rated up to 90% and above, Medium are rated less than 10% respectively.

The Relevance of the system is calculated thus:

$$\text{Relevance} = \frac{\sum_{i=1}^{N=4} k_i * r_i}{N * \sum_{i=1}^{N=4} k_i},$$

where N is the total number of rate point, $r = 1, \dots, N$ and k_i is the sum of user that selected a given rate point for all the metrics.

$$\begin{aligned} &= \frac{(680 \times 4) + (929 \times 3) + (93 \times 2) + (0 \times 1)}{1702 \times 4} \\ &= \frac{2720 + 2787 + 186}{6808} = \frac{5693}{6808} \\ &= 0.8362 = 83.62\% \end{aligned}$$

5. Conclusion and Future Works

An answer validation system for answers using answerers attributes and crowd ranking has been developed. For the effectiveness of the system, illegitimate questions and answers were filtered out using a trained Naïve Bayes spam filter with a threshold of 0.5. Answerers’ personal attributes (such as Grade points, Area of specialization, Years of experience Level of Computing, Course of study, Question Understandability and the answer confidence level (trustworthiness)) were used to ensure high quality answers by employing a weighted system that assigns weights to individual attributes in order to know the weight of the answers for validation. Answers are ranked by the crowd to get the best four answers from the candidate answers obtained from the answerers using Borda count ranking algorithm and least best answer is discarded. The system correctness is 96.47%, Answer satisfaction is 100%, answer Validation is 97.65%, system Simplicity is 97.6%, system Feedback is 88.23% and the system efficiency is 96.47%. Future works could include more User attributes such as age, qualification and so on and ensure that there is an improvement in the system feedback so that users can receive instant live answers to their respective questions. There should be a way in which the answerers are motivated for the task performed in order to enhance their performance. In addition, the system should be more general to accommodate questions from other science related domain.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

Anderson, A., Huttenlocher, D., Kleinberg, J., & Leskovec, J. (2012). Discovering Value

- from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overflow. *Proceedings the 18th ACM International Conference on Knowledge Discovery and Data Mining*, Beijing, 12-16 August 2012, 850-858. <https://doi.org/10.1145/2339530.2339665>
- Aydin, B., Yilmaz, Y., Li, Y., Li, Q., Gao, J., & Demirbas, M. (2014). Crowdsourcing for Multiple-Choice Question Answering. *Proceedings the 26th Annual Conference on Innovative Applications of Artificial Intelligence*, Québec, 29-31 July 2014, 1-12.
- Chandra, A., Reddy, O., & Madhavi, K. (2017). A Survey on Types of Question Answering System. *IOSR Journal of Computer Engineering*, 19, 19-23.
- Cimiano, P., Unger, C., & McCrae, J. (2014). *Ontology-Based Interpretation of Natural Language*. San Rafael, CA: Morgan & Claypool Publishers. <https://doi.org/10.2200/S00561ED1V01Y201401HLT024>
- Dobšovič, R., Grznar, M., Harinek, J., Molnar, S., Palenik, P., Poizl, D., & Zbell, P. (2014). *Askalot: An Educational Community Question Answering System*. Unpublished PhD Thesis, Bratislava: Slovak University of Technology.
- Fan, H., Ma, Z., Li, H., Wang, D., & Liu, J. (2019). Enhanced Answer Selection in CQA Using Multi-Dimensional Features Combination. *Tsinghua Science and Technology*, 24, 346-359. <https://doi.org/10.26599/TST.2018.9010050>
- Harper, F., Raban, D., Rafaei, S., & Konstan, J. (2008). Predictors of Answer Quality in Online Q&A Sites. *Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, Florence, 5-10 April 2008, 865-874. <https://doi.org/10.1145/1357054.1357191>
- Howe, J. (2006). *Crowdsourcing: A Definition*. Wired Blog Network: Crowdsourcing. http://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_a.html
- Hung, N., Thang, D., Tam, N., Weidlich, M., Aberer, K., Yin, H., & Zhou, X. (2017). Answer Validation for Generic Crowdsourcing Tasks with Minimal Efforts. *The VLDB Journal*, 26, 855-880. <https://doi.org/10.1007/s00778-017-0484-3>
- Hung, N., Thang, D., Weidlich, M., & Aberer, K. (2015). Minimizing Efforts in Validating Crowd Answers. *Proceedings of the Association for Computer Machinery's Special Interest Group on Management of Data*, Melbourne, May 2015, 999-1014. <https://doi.org/10.1145/2723372.2723731>
- Ishikawa, D., Kando, N., & Sakai, T. (2011). What Makes a Good Answer in Community Question Answering? An Analysis of Assessors' Criteria. *The Fourth International Workshop on Evaluating Information Access*, Tokyo, December 2011, 169-181.
- Le, L., Shah, C., & Choi, E. (2019). Assessing the Quality of Answers Autonomously in Community Question-Answering. *International Journal on Digital Libraries*, 20, 1-17.
- Magnini, B., Negri, M., Prevete, R., & Tanev, H. (2002). Comparing Statistical and Content-Based Techniques for Answer Validation on the Web. *Proceedings of the 8th Convegno AI&IA*, Siena, September 2002, 413-427.
- Magnini, B., Negri, M., Prevete, R., & Tanev, H. (2005). Is It the Right Answer? Exploiting Web Redundancy for Answer Validation. *ACL-02: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, July, 425-432.
- Nie, L., Wei, X., Zhang, D., Wang, X., Gao, Z., & Yang, Y. (2017). Data-Driven Answer Selection in Community QA Systems. *IEEE Transactions on Knowledge and Data Engineering*, 29, 1186-1198. <https://doi.org/10.1109/TKDE.2017.2669982>
- Ojokoh, B., & Adebisi, E. (2019). A Review of Question Answering Systems. *Journal of Web Engineering*, 17, 717-758. <https://doi.org/10.13052/jwe1540-9589.1785>
- Ojokoh, B., & Ayokunle, P. (2012). Fuzzy-Based Answer Ranking in Question Answering

- Communities. *International Journal of Digital Library Systems*, 3, 47-63.
<https://doi.org/10.4018/jdls.2012070105>
- Ríos-Gaona, M., Gelbukh, A., & Bandyopadhyay, S. (2012). Recognizing Textual Entailment Using a Machine Learning Approach. In *Mexican International Conference on Artificial Intelligence* (pp. 177-185). Berlin: Springer.
https://doi.org/10.1007/978-3-642-16773-7_15
- Savenkov, D., Weitzner, S., & Agichtein, E. (2016). Crowdsourcing for (Almost) Real-Time Question Answering. *NAACL 2016: Proceedings of the Workshop on Human-Computer Question Answering*, San-Diego, 12-17 June 2016, 8-14.
<https://doi.org/10.18653/v1/W16-0102>
- Schofield, M., & Thielscher, M. (2019). General Game Playing with Imperfect Information. *Artificial Intelligence Journal*, 66, 901-935. <https://doi.org/10.1613/jair.1.11844>
- Šimko, J., Simko, M., Bielíková, M., Sevech, J., & Burger, R. (2013). Classsourcing: Crowd-Based Validation of Question-Answer Learning Objects. In *International Conference on Computational Collective Intelligence* (pp. 62-71). Berlin: Springer.
https://doi.org/10.1007/978-3-642-40495-5_7
- Su, Q., Pavlov, D., Chow, J., & Baker, W. (2007). Internet-Scale Collection of Human-Reviewed Data. *Proceedings of the 16th International Conference on World Wide Web*, Banff, 8-12 May 2007, 231-240. <https://doi.org/10.1145/1242572.1242604>
- Tan, J., Rönkkö, K., & Gencel, C. (2010). *A Framework for Software Usability and User Experience Measurement in Mobile Industry*. MSc Thesis, Karlshamn: Blekinge Institute of Technology, Sweden.
- Toba, H., Ming, Z., Adriani, M., & Chua, T. (2014). Discovering High Quality Answers in Community Question Answering Archives Using a Hierarchy of Classifiers. *Information Sciences*, 261, 101-115. <https://doi.org/10.1016/j.ins.2013.10.030>
- Tran, Q., Tran, V., Vu, T., Nguyen, M., & Pham, S. (2015). JAIST: Combining Multiple Features for Answer Selection in Community Question Answering. *Proceedings of the 9th International Workshop on Semantic Evaluation*, Denver, 4-5 June 2015, 215-219.
<https://doi.org/10.18653/v1/S15-2038>