IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 September 2022, accepted 29 September 2022, date of publication 25 October 2022,
date of current version 21 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3216840

RESEARCH ARTICLE

Blockchain-Based Software Effort Estimation: An Empirical Study

MANSOOR AHMED[1,2], NAEEM IQBAL[3], (Member, IEEE), FARAZ HUSSAIN[2], MURAD-ALI KHAN[3], MARKUS HELFERT[1], IMRAN[4], AND JUNGSUK KIM[4,5], (Member, IEEE)

[1]ADAPT Centre, Innovation Value Institute, Maynooth University, Maynooth, W23 A3HY Ireland
[2]Department of Computer Science, COMSATS University, Islamabad, Federal Capital 44000, Pakistan
[3]Department of Computer Engineering, Jeju National University, Jeju 63243, Republic of Korea
[4]Department of Biomedical Engineering, Gachon University, Incheon 21936, Republic of Korea
[5]Research and Development Laboratory, Cellico Company, Seongnam-si, Gyeonggi-do 13449, Republic of Korea

Corresponding authors: Imran (imranj@gachon.ac.kr) and Jungsuk Kim (jungsuk@gachon.ac.kr)

This work was supported by the National Research Foundation of Korea under Grant NRF-2022R1A2C1012037.

ABSTRACT Context: The success or failure of any software development project significantly depends on the accuracy of its effort estimates. Software development effort estimation is the foundation for project bidding, budgeting, planning, and cost control. Problem: The literature shows that a lot of work has been done on software effort estimation. But still, there is a need for improvement in effort estimation by introducing new methodologies. The structured group-based and analogy-based effort estimations are the widely used estimation methods. Nevertheless, there are several shortcomings of using these methods such as lack of experts, lack of historical data, and biasness in expert opinion, which negatively affect the estimation results. Motivation: With the advancement of technologies, such limitations could be overcome. Such as leveraging the applicability of blockchain in several domains such as improvement in the software development process and network security. Method: In this article, we have proposed a Blockchain-Based Software Effort Estimation (BBSEE) methodology to improve the software effort estimation. We employ the proposed method using Web and blockchain technologies. Moreover, we also proposed evaluation criteria to assess the efficacy of the proposed method in terms of Mean Magnitude of Relative Error (MMRE), Mean Absolute Error (MAE) and percentage of successful predictions falling (PRED (25)). Result: We performed several case studies and analyses of expert opinions of 52 organizations to present the efficacy of the proposed method. Conclusion: We observe that the BBSEE method outperforms expert judgment and analogy-based effort estimation methodologies in terms of software effort estimation.

INDEX TERMS Software effort estimation, blockchain, blockchain-based software engineering, analogy-based estimation, group-based estimation, estimation error, software engineering.

I. INTRODUCTION

The software development effort estimation is the measure of most likely effort [1]. Effort estimation is a crucial software management activity and provides a foundation for project planning, budgeting, bidding, and cost control [2]. Although, a lot of work is done on software effort estimation [3], however, literature shows a gap of improvement in the effort estimation methodologies, especially in the context of agile software development where requirement changes frequently [4], [5]. Moreover, the success or failure of any software project significantly depends on the accuracy of its effort estimation as indicated by the analysis of the Standish group [6]. In the era 2011-2015, the analysts have determined the success and the failure rate of more than 25000 projects in the era of 2011-2015. According to their

The associate editor coordinating the review of this manuscript and approving it for publication was Porfirio Tramontana.

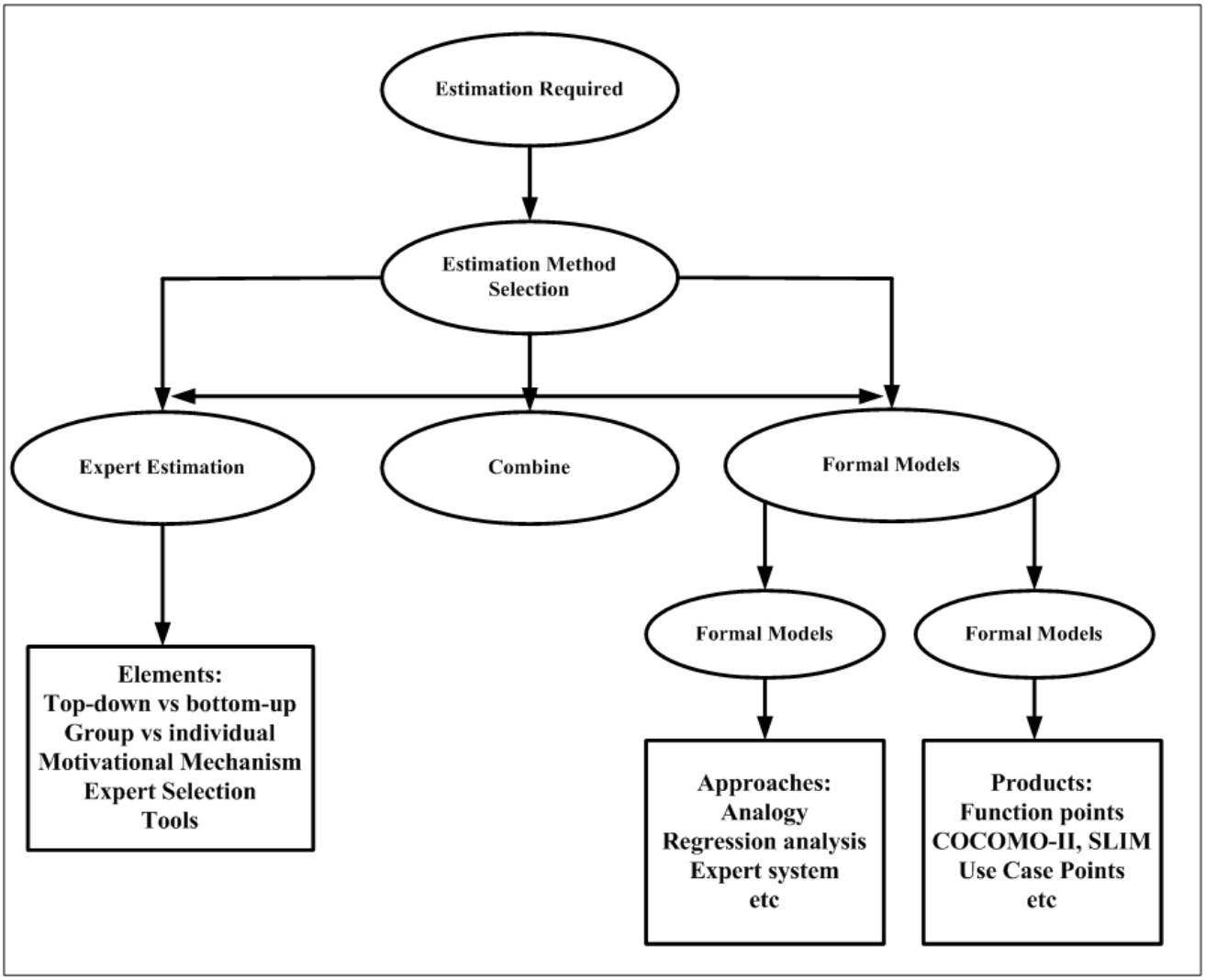

FIGURE 1. The layout of existing software estimation methodologies.

analysis, 27-31% of the projects were successful (on budget, on time, acceptable), 17-22% of the projects failed (either canceled or rejected on delivery), and 4956% of the projects were highly challenged (over budget and schedule but rated satisfactory). Moreover, the annual average budget overrun was 68-72% and the annual average schedule overrun was 66-81% [6]. The level of overrun in budget and schedule arises a need for improvement in existing software effort estimation methodologies. The common estimation methods used can be seen in the decision tree shown in Figure 1, amongst the methods Expert judgment is the most used approach [1]. The layout of existing methodologies is shown in Figure 1, which indicates that structured group-based effort estimation methods are widely used [7]. Although, there are several benefits of using group-based effort estimation techniques reported. However, there are numerous shortcomings of their use also reported such as lack of experts and biasness in expert opinion [8], [9]. Moreover, analogy-based methods are the other most common effort estimation tool used by industry experts [7]. However, it also has some limitations which include a lack of historical data, especially in small organizations [8]. Even though the research community is very active and has proposed many estimation methods, the results are not substantial as observed in the analysis of the Standish group [6]. Moreover, finding domain-specific expert opinions and project-relevant data for precise estimation remains always a dream of a project manager. The availability of experts and reliable historical data always lead to the success of a project estimate. Subsequently, the current situation encouraged us to propose a method that would overcome the above-discussed challenges and ultimately improve the effort estimation results. The revolution in the advancement of technology is taking place very rapidly and opening new directions to solve different problems. Besides, the researchers are strongly motivated to adopt advanced technologies to solve the trending effort estimation problem, such

as Machine Learning (ML) based estimation methods introduced to improve the accuracy of software effort estimation to encounter the shortcomings of existing techniques [10].

From the list of recent technological techniques, Software Engineering (SE) and Computer Network (CN) community has started to leverage the capabilities of blockchain to improve its performance. Blockchain is a distributed and decentralized ledger of transactions, it has also been acknowledged as a digital currency platform after introducing bitcoin the largest cryptocurrency [11]. The applications of blockchain are gradually increasing and a lot of research has been carried out. In the past few years blockchain was introduced to the domain of computer science to solve several problems [12], such as the implications of blockchain to improve the software process [36]. Recently, the blockchain was also introduced to the domain of the SE research community and software engineers are focusing on blockchain implications to resolve certain software development issues [13], [14]. This gives us a new direction to solve the software effort estimation inaccuracy problem using blockchain. In our opinion, the use of a blockchain-based method, that integrates the benefits of Group and analogy-based estimation can overcome their limitations that are why we proposed a blockchain-based solution. The main contributions of the proposed BBSEE can be summarized as follows.

- The proposed BBSEE overcomes the problem of a lack of experts by introducing a remote network that encapsulates a verifiable community of estimation experts that are always available over the network.
- The proposed BBSEE overcomes the limitation of biasness in the physical existing group by introducing a remote network that encapsulates a verifiable community of estimation experts, whose decision is not affected by any type of biasness.
- The proposed BBSEE overcomes the problem of lack of historical data by introducing a blockchain-based repository, where historical data is securely stored and publicly available to be used over the network.

The rest of this article is structured as follows. Section II defines the related studies and a summary of findings. The commonly used estimation methods were described in Section III. In Section IV, the proposed methodology is presented. In Section V, the research design is defined. Section VI presents the results and discussion of this study. Furthermore, the threats to validity are presented in Section VII. In Section VIII, the conclusion and future work of the proposed research are defined.

II. RELATED WORK

The success rate of software projects is challenging, from the analysis of the Standish group, we came across the fact that the success of any software project is significantly dependent on the accuracy of its effort estimates. Several studies have been examined and contributed to the improvement of effort estimation, but the level of improvement was not satisfactory [6].

Most of the work has been done on effort estimation which focused on proposing and improving the traditional models. One of these models is the COCOMO model which is introduced by Barry Boehm. The COCOMO model is continuously improved with time and gained more attention from the SE research community [15], [16], Similarly, Function Point-based analysis was introduced by Albrecht to measure the size of the functionality of systems [17]. Similarly, an analogy-based generic method is introduced for effort estimation, whose main idea was that whenever a new project is started, the historical data of an existing similar project is viewed and used while making estimation decisions [18]. Similarly, many other traditional methods were proposed and improved continuously, but there was no better effort estimation method [9], [19].

As agile software development is achieving popularity gradually and replacing traditional methods of software development [20]. Different studies have acknowledged that the accuracy of software effort estimation is more crucial in the context of agile software development [4], [5]. For instance, technology is evolving with each elapsing day and exploring new dimensions to solve different problems, due to which the research community is keenly interested to adopt advanced technologies to solve the problem of inaccuracy in effort estimation, Furthermore, the literature review indicates that several methods have been proposed [19], such as

Ontology Oriented Software Effort Estimation (OOSEE) was introduced for E-commerce applications based on Extreme programming, and Scrum methodologies and an Ontology-based system for the agile method was also introduced. The common aim of these techniques was to improve the accuracy of effort estimation [21], [22]. Similarly, certain machine learning-based effort estimation methods are also proposed. The study of Polkowski et al. [23] states that machine learning-based effort estimation increases estimation accuracy [23]. Systematic Literature Review (SLR) compared different machine learning-based effort estimation methods so that one might select appropriate and effective effort estimation methods for the desired context [24]. Although, Machine Learning (ML) based effort estimation methods are comprehensive and acknowledged extensively. However, there are some limitations to using them, as indicated by Shukla et al. The ML-based methods do not work well on all datasets. Moreover, this paper recommends exploring the Neural Networks-based estimation method to improve estimation accuracy [10].

The use of Neural Networks in estimation is a well-practiced approach as indicated by Nassif et al [25]. In this article, the authors have compared different Neural Networks-based methods [25] and reported the outperformed method. Similarly, in research, it is specified that the creative and abstract nature of software makes it difficult to measure its estimation accuracy and the use of neural networks can help in the improvement of estimation accuracy [26].

Similarly, the use of Neural Networks also encounters some shortcomings [19]. The emerging situation needs a perfect matching technology that can improve the estimation results. Finally, SE researchers have proposed several approaches which are exceptionally capable to manage the uncertainty as effort, duration, and velocity prediction [3].

Blockchain applications are increasing gradually which is absorbed in a paper by Porru et al. [13]. In this article, the authors have acknowledged that there is a need for blockchain-based software development [13]. Similarly, in a concept paper by Beller et al. [14], the authors published and formulated the term ''Blockchain-based Software Engineering'' in which they introduced the concept of solving software engineering problems using Blockchain.

The divergence of the research community toward the use of advanced technologies has motivated us to solve the software effort estimation inaccuracy problem using Blockchain. To the best of our knowledge, no work is previously done on Blockchain-based software effort estimation.

According to a survey [7], the most frequently used estimation methods such as structured group and analogy-based effort estimation are reported. But these methods do not give significant results due to their above-discussed limitations. The shortcomings of group-based effort estimation include lack of experts and biasness in expert opinion [8], [9], Furthermore, in analogy-based estimation, the accuracy of estimates significantly depends on the availability of historical data [8]. This study proposed a blockchain-based solution that will overcome the above-mentioned limitations and improve the estimation results.

III. ESTIMATION BY ANALOGY AND EXPERTS GROUP

There are different estimation methods available (as shown in Figure 1), from which you can select and use the most suitable one, to get better results. From Figure 1, we can observe that there are three options available to perform effort estimation, such as Expert estimation, Formal models, or a combination of both. Most of the time, combined approaches are used to get better results [27]. We have further selected two methods based on their popularity, which are Expert group and Analogy-based estimation, and preferred to use them in combination.

A. GROUP ESTIMATION

The usage of combining effort estimates on average leads to more accurate estimates, especially in the case where it is not clear which method is better or who has the relevant experience, the essential technique is that estimation should be done independently from each other and takes benefit from their combine usage [9]. There are two ways to arrange a group, one of them is known as a structured group and the other is known as an unstructured group. In effort estimation, the use of an unstructured group is not appreciated by the experts. The experts recommend using a structured group arrangement while estimating, there are different techniques based on the structured group. The most common techniques include Planning poker and Wide-band Delphi, the high-level working of wide-band Delphi is given as [28].

1) GETTING READY FOR THE ESTIMATION PROCESS

- Develop estimation-related supporting material
- Finalize estimation team and their moderator

2) START THE ESTIMATION MEETING

- Moderator briefs the team about the estimation problem, estimation stuff, estimation process, etc.
- Team discussion of the members of the team and supporting material

3) ESTIMATION BY EACH TEAM MEMBER INDIVIDUALLY

- Identification of activities and estimates
- Experts that are not part of the estimating team can be consulted

4) ESTIMATION MEETING

- Moderator briefs the team about the estimates received
- All Experts discuss the estimates

5) SUM UP THE ESTIMATION PHASE

- Done by moderator and project leader

B. ANALOGOUS ESTIMATION

The most common example of analogy-based project estimation is case-based reasoning, where identical projects from the lesson learned are identified and used for estimation. They can be used in combination with collective expert opinion and formal models [29], Software effort estimation by using an analogy-based tech commonly involves the following steps.

1) Measuring the values of identified metrics of the software project for which estimation is being performed (target project).
2) Finding a similar project from the repository.
3) Using the estimated effort values of the selected projects to use as an initial estimate for the target project.
4) Comparison of metric's value for the target project and selected project.
5) Adjustment of effort estimates in view of the comparison performed in the previous step.

IV. PROPOSED BBSEE METHOD

A. BBSEE OVERVIEW

The proposed method BBSEE is a concept of the freelance effort estimation market. It overcomes the limitations of group-based estimation and analogy-based estimation by using the blockchain concept. The BBSEE helps organizations lacking the estimation abilities to estimate and improve their estimated results.

B. BBSEE PROCESS FLOW

Our proposed Blockchain-Based Software Effort Estimation (BBSEE) method comprises four layers, namely the

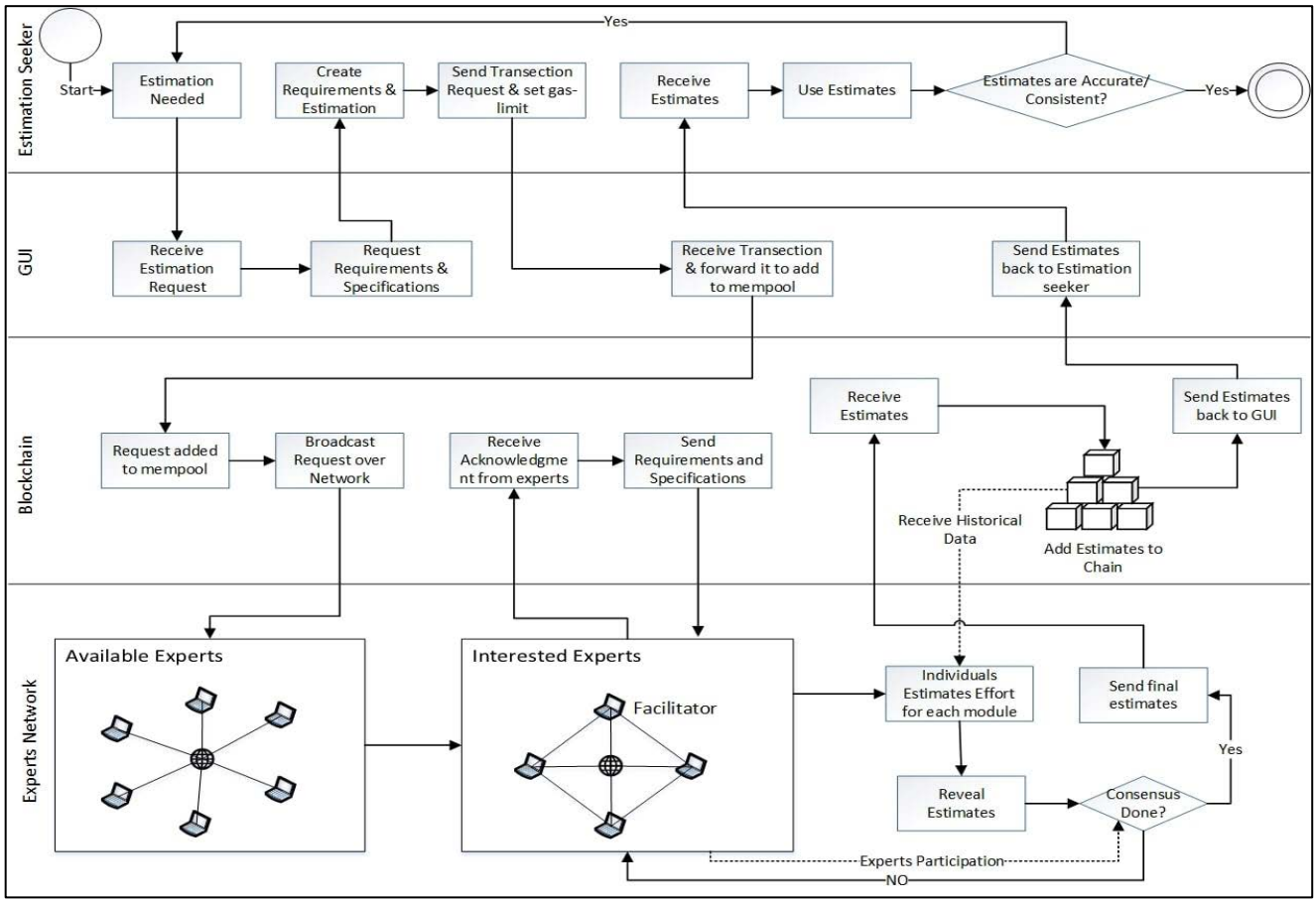

FIGURE 2. BBSEE workflow.

Estimation seeker layer, Graphical User Interface layer (GUI), Blockchain layer, and Experts network layer.

The proposed BBSEE method is shown in Figure 2, which contains a network that encapsulates a verifiable community of estimation experts that carry out effort estimation for organizations and individuals. The effort estimation with the proposed BBSEE requires the estimation seeker from the estimation seeker layer to send an estimation request to the GUI layer, which will send back the required template for the requirement and specifications of the system. The seeker is supposed to write requirements and specifications in the defined template and send it along with the transaction request to the GUI layer. Since the fixed number of activities are performed in both layers. So, the runtime complexity of both layers could be computed as O(c). The template of the block is used to group the information into two sections, namely header and transaction sections. The header section of the block contains information about hash reference with another block and timestamp. While the transaction section contains information related to requirements and specifications of cost estimations retrieved from the experts. Afterward, the transaction is forwarded to the blockchain layer and added to the mempool (waiting queue for transactions), and upon its turn estimation request broadcasts to the network of available experts on the expert's network layer, from which the interested experts accept the request. Due to the involvement of n transactions, the runtime complexity of the blockchain layer will remain O(n). Subsequently, it is added to a network of selected experts that will carry out estimation decisions for that project, from the list of selected experts, one of the experts is selected as a facilitator on the base of their experience. Due to the involvement of the expert network, the complexity of the ''Expert Network'' layer is O(V∗E), where V referred to the list of experts while E referred to the values to describe the relationship between experts.

Afterward according to requirements and specifications each expert individually divides the software system into modules and estimates the effort required for each module using historical data available over the chain of the Blockchain layer. Each expert reveals its estimates, a joint consensus is done by all the experts, where the facilitator helps them in reaching the final estimate. The facilitator forwards the final estimates to the blockchain layer. Subsequently, the final estimates are then added to the chain for future use. Moreover, the estimates are sent back to the GUI layer which passes them to the Estimation seeker layer

where the seeker receives the estimates. The final estimates are used by the estimation seeker, who continuously checks for estimation accuracy on the base of which decision is made, whether to use the estimates or re-estimate again for improvement in estimates accuracy. The detailed architecture of the proposed BBSEE method is shown in Figure 2. The overall complexity of BBSEE is computed as follows.

$$O(BBSEE) = O(O(c) + O(c) + O(n) + O(V \times E)) = O(V \times E)$$

C. BBSEE BENEFITS

The combination of different effort estimation methods helps to achieve the benefits of multiple estimation methodologies. Our proposed system overcomes the challenges such as lack of experts, biasness in expert opinion, and lack of historical data by combining the benefits of group-based and analogy-based effort estimation using blockchain. As a result, this approach would help to improve the estimation accuracy.

V. RESEARCH DESIGN

We have selected the software effort estimation domain and performed an exploratory study to identify the most used practices and their limitations. The results of the exploratory study identified that Analogy and Group-based estimation methods are the most used estimation practices. We have explored a solution that can overcome their limitations, namely Blockchain-Based Software Engineering (BBSE), which gives us an idea to propose a new method. We have proposed the Blockchain-Based Software Effort Estimation (BBSEE) methodology, which overcomes the limitations of group and analogy-based effort estimation. For the empirical evaluation of the BBSEE, we have simulated all its possible cases and developed a working BBSEE system. Afterward, two case studies were performed to evaluate current practices and collect feedback about the BBSEE. Similarly, the performance of the proposed BBSEE is also evaluated using two experiments in which the performance measures MMRE, MAE, and PRED (25) are used. The results of the case studies and experiments were collected and analyzed to prove our claims; the overview of the research can also be seen in Figure 3.

A. BBSEE SIMULATION

All the evaluations required in this research (Case study I, Case study II and performance evaluation) need the availability of a working BBSEE system, this brings us to develop a Decentralized Application (dApp), which covers all the workflows of the proposed methodology as shown in Figure 2. The proposed system was developed using solidity, JavaScript, PHP, MySQL, and HTML. and CSS. The system was deployed using 000webhost, and the firebase and Rinkebey testnet were used to automate the file storage, retrieval, and execution of smart contracts, and to make sure that the file is secure, decentralized, and publically accessible to make the system work as intended.

B. CASE STUDY I

1) PARTICIPANTS

We invited over 120 software organizations to participate in the activity from which only 76 organizations express interest to participate, amongst which only 52 organization representatives participated in the activity. All the representatives have their academic background in computer science and have professional experience in software development, some of the participants also had blockchain backgrounds and most of the representatives were serving their organizations in senior positions.

2) PROCEDURE

This study aims to analyze the current situation of organizations. Our main aim was to evaluate the ratio of the organizations performing effort estimation activity. The ratio of the organizations conforming to the standard effort estimation methodology and identifying the most frequently used estimation methods and their major barriers that prevent organizations from performing estimation activities. We evaluated this study by performing a short presentation session. We have presented a brief overview of the existing effort estimation methodology and asked the participants to answer a set of questions regarding the current practices of their organizations.

C. CASE STUDY II

1) PARTICIPANTS

The participants of case study II were the same 52 representatives of the organization who participated in case study I.

2) PROCEDURE

The second case study is performed to evaluate and validate the proposed BBSEE. The major goals of the proposed system are to overcome the above-identified limitations and to make the estimation activities efficient and convenient for all organizations. We perform this study by providing a brief overview of the traditional methods (along with their shortcomings), and the benefits of the proposed BBSEE method were presented using a storyboard. To make the audience familiar with our proposed system, we have presented a demonstration of the use of the proposed system. We elaborated on the above-mentioned limitations and requested the audience to use the proposed BBSEE method and give their feedback about it.

D. PERFORMANCE EVALUATION

Prediction accuracy is an important part of the estimation method; we have performed two experiments, in which we have selected 6 experts who were currently working in the software industry with 3-6 years of professional experience. In the first experiment, the experts use structured group-based estimation (Wide-band Delphi) for performing estimation activities. In the second experiment, the experts use the proposed BBSEE method for performing estimation activities.

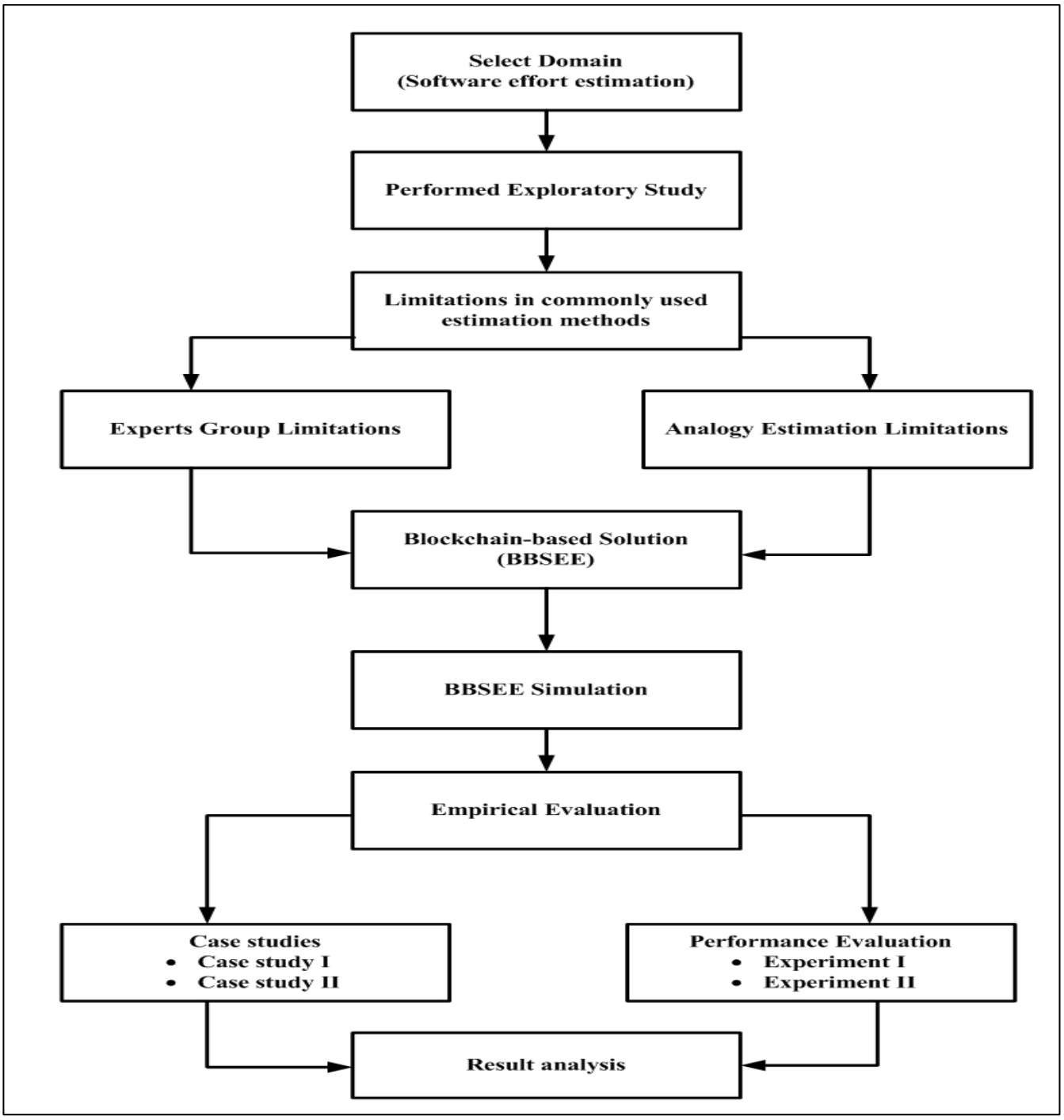

FIGURE 3. Research design.

1) EXPERIMENT

From the list of 81 projects, we have selected 15 projects for estimation using Wide-band Delphi, selected projects can be viewed in Table 11.

We have selected 15 projects from the list of 81 projects and instructed our experts to use the Wide-band Delphi technique to estimate all the 15 selected projects one by one and monitored them while they were performing estimation activities. The results of the estimation were used for evaluating the performance of the Wide-band Delphi.

2) EXPERIMENT II

We have selected other 15 projects from the dataset of 81 projects and used the remaining 66 projects as

historical data and introduced the selected experts to our proposed BBSEE method. Afterward, we performed a controlled experiment in which the selected experts were asked to estimate the target 15 projects using the proposed BBSEE method. The results of the experiment were used to evaluate the performance of BBSEE.

3) EVALUATION MEASURES

The commonly used evaluation criteria for effort estimation methods are the Magnitude of Relative Error (MRE) [30], which is represented in equation 1.

$$MRE = \frac{Actual\ Effort - Estimated\ Effort}{Actual\ Effort} \tag{1}$$

The MRE is calculated for all selected N projects, and the Mean Magnitude of Relative Error (MMRE) is the average of N projects [31], as defined in equation 2.

$$MMRE = \frac{1}{N}\sum_{i=1}^{N} MRE_i \tag{2}$$

The success percentage of predictions laying within 25% of the actual values is also represented as PRED(25) and defined in equation 3, where N is the number of selected projects [32].

$$PRED(25) = \frac{100}{N}\sum_{i=1}^{N} \begin{cases} 1 & if\ MRE_i \leq \frac{25}{100} \\ 0 & otherwise \end{cases} \tag{3}$$

The Mean Absolute Error (MAE) is the average magnitude of error in a set of predictions as defined in equation 4, where y_i is prediction, x_i is the actual value and n is the total number of points.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{4}$$

4) DATASET

The dataset used in experiments I & II comes from the Desharnais promise repository dataset [33]. We have collected data on 81 projects. The detailed statistics for the dataset can be viewed using the promise repository.

VI. RESULTS AND DISCUSSION

This section reports and evaluates the results of two case studies in which the representatives of 52 organizations participated. This section also reports the results of two experiments in which effort estimation for the selected project was performed using BBSEE and Wide-band Delphi to evaluate their performance.

A. CASE STUDY I

The first case study aimed to analyze the current practices of organizations. Our main aims of this case study include evaluating the current practices, extracting demographic information, and identifying the barriers preventing organizations from performing estimation activities. We evaluated the study by performing a short presentation session. In this section, we have considered a few hypotheses to understand the developer's opinion about cost estimation practices in terms of their self and their organization perspectives.

1) HYPOTHESIS-1

Software firms are performing some sort of software effort estimation practices. According to the response of the experts' 63.5% of the organizations perform estimation and somehow 36.5% of the organization doesn't perform the estimation. The results imply that the majority of the organizations perform effort estimation as shown in Table 1.

TABLE 1. Organizations performing effort estimation.

Response	Respondents	Total	%
Yes	33	52	63.5
No	19	52	36.5

2) HYPOTHESIS-2

Software firms are using well-known methodologies for software effort estimation.

The results of Table 2 indicate that 81.8% of organizations were conforming to standard estimation methodologies, whereas 18.2% of organizations don't conform to any effort estimation methodology..

TABLE 2. Organizations using estimation methods.

Response	Respondents	Total	%
Yes	27	33	81.8
No	6	33	18.2

3) HYPOTHESIS-3

Software firms are more inclined toward the Group-based estimation process.

The results of the Table 3 indicate that most organization is following a group-based estimation rather than following some standards such as COCOMO Model.

Hence, it can be concluded that the most frequently used effort estimation methods are group-based and analogy-based effort estimation methods.

TABLE 3. List of estimation methods.

Response	Respondents	Total	%
Group-based estimation	24	27	88.8
Analogy estimation	11	27	40.7
Use-case point	8	27	29.6
COCOMO model	4	27	14.8
function point	3	27	11.1
individual estimation	3	27	11.1
other methods	1	27	3.7

4) HYPOTHESIS-4

Software firms are facing barriers to performing effort estimation activities. The results of Table 4 indicate that there are several barriers for the organization to perform precise effort estimation such as Lack of Experts, Time, group biasness, and lack of historical data. Moreover, we can observe that lack of experts and historical data are the major barriers to an organization.

TABLE 4. List of estimation barriers.

Response	Respondents	Total	%
lack of historical data	11	27	40.7
lack of experts & cost	9	27	33.3
Group biasness	6	27	22.2
resource management	4	27	14.8

In the case study, I was aiming to extract the current practices of participating organizations, the results of the study favor our claims that the most frequently used estimation methods were group-based and analogy-based estimation. The results of the study also favor our claim that lack of experts, lack of historical data, and biasness in the group were the major barriers preventing organizations from performing estimation activities. The results of the study also identified that the majority of organizations were performing effort estimation and following different techniques for estimation.

B. CASE STUDY II

The second case study is performed to assess the efficacy of the proposed BBSEE. We performed this case study with experts who have experience with BBSEE and collected their feedback to assess our hypotheses, such as.

1) HYPOTHESIS-5

BBSEE could aid organizations to overcome the problem of lack of experts.

The result of Table 5 indicates that 77% of expert agrees with to use of the BBSEE method to improve the effort estimation in case of a lack of experts in the organization. Moreover, only 5% are not agreeing to the use of BBSEE when organizations don't have experts in effort estimations.

TABLE 5. BBSEE feedback to overcome the lack of experts.

Response	Respondents	Total	%
Strongly Agree	11	52	21.2
Agree	29	52	55.8
Neutral	7	52	13.5
Disagree	5	52	9.6
Strongly Disagree	0	0	0

2) HYPOTHESIS-6

BBSEE could aid organizations to overcome the problem of the expert's biasness.

The result of Table 6 indicates that 73.1% of experts agree with to use of the BBSEE method to improve the effort estimation in terms of overcoming the expert biasness. Moreover, only 1% is not agreeing with the use of BBSEE to overcome the expert biasness.

TABLE 6. BBSEE feedback to overcome biasedness in the group.

Response	Respondents	Total	%
Strongly Agree	14	52	26.9
Agree	24	52	46.2
Neutral	13	52	25
Disagree	1	52	1.9
Strongly Disagree	0	52	0

3) HYPOTHESIS-7

BBSEE could aid organizations to overcome the problem of lack of experts.

The result of Table 7 indicates that 70.1% of expert agrees with to use of the BBSEE method to improve the effort estimation in case of a lack of historical data. Moreover, only 9.6% are not agreeing with the use of BBSEE when an organization doesn't have historical data for effort estimations.

TABLE 7. BBSEE feedback to overcome the lack of historical data.

Response	Respondents	Total	%
Strongly Agree	10	52	19.2
Agree	32	52	61.5
Neutral	5	52	9.6
Disagree	4	52	7.7
Strongly Disagree	1	52	1.9

4) HYPOTHESIS-8

BBSEE freelance market could affect the overall cost of effort estimation.

The result of Table 8 indicates that 52.9% of experts agree with the significant decrease in the cost of effort estimation by incorporating the use of the BBSEE freelance market. Moreover, only 28.9% do not agree with the decrease in the cost of effort estimation using the BBSEE freelance market.

TABLE 8. BBSEE effect on cost.

Response	Respondents	Total	%
Significantly Decreased	17	52	32.7
Minorly Decreased	10	52	19.2
Neutral	10	52	19.2
Minorly Increased	8	52	15.4
Significantly Increased	7	52	13.5

5) HYPOTHESIS-9

BBSEE freelance market could affect the overall duration of effort estimation.

The result of Table 9 indicates that 51.9% of expert agrees with the significant decrease in duration of effort estimation by incorporating the use of the BBSEE freelance market. Moreover, only 28.8% do not agree with the decrease in duration of effort estimation using the BBSEE freelance market.

TABLE 9. BBSEE effect on time.

Response	Respondents	Total	%
Significantly Decreased	12	52	23.1
Minorly Decreased	15	52	28.8
Neutral	10	52	19.2
Minorly Increased	9	52	17.3
Significantly Increased	6	52	11.5

6) HYPOTHESIS-10

BBSEE method can perform effort estimation activities more efficiently as compared to traditional methods.

The result of Table 10 indicates that 80.8% of expert agrees that the BBSEE method can perform effort estimation activities more effectively as compared to traditional methods. Moreover, only 19.2% do not agree that effort estimation activities are not performed efficiently as compared to the existing methods.

TABLE 10. BBSEE and traditional methods of feedback.

Response	Respondents	Total	%
Strongly Agree	13	52	25.0
Agree	29	52	55.8
Neutral	8	52	15.4
Disagree	2	52	15.4
Strongly Disagree	0	52	0

Case study II aimed to assess the proposed BBSEE and validate it from the industry, and the results of the study are in favor of our assumptions about the proposed BBSEE. Moreover, the results support our claim that the use of BBSEE can overcome the problems of lack of expertise, lack of historical data, and biasness in group decisions. We can also imply from the results that the use of the proposed BBSEE will decrease the cost and time required for estimation activities. The results also imply that the proposed BBSEE method could efficiently perform estimation activities as compared to existing traditional models.

C. PERFORMANCE EVALUATION

1) EXPERIMENT I

The expert performed the estimation of the selected 15 projects using the Wide-band Delphi technique, the estimation and performance evaluation results can be viewed in Table 11.

The performance evaluation results of the experiment I report that MMRE for Wide-band Delphi is 0.256, MAE is 1049, and PRED (25) is 60%. This implies that the performance of the Wide-band Delphi technique is not satisfactory. The software estimation method is considered satisfactory if its MMRE is less than 0.25 and PRED (25) greater than 75% [34].

We implied that the performance of the Wide-band Delphi is not satisfactory due to the shortcoming of group-based estimation. As discussed earlier the lack of relevant experts and biasness in the group affect the estimation results negatively, additionally the availability of historical data and lack of historical data while making estimation decisions can also improve the estimation results.

TABLE 11. Wide-band Delphi performance.

Project	Actual	Estimated	MRE	PRED(25)
1	3948	5200	0.317	0
2	3927	3100	0.210	1
3	710	900	0.267	0
4	2429	2900	0.193	1
5	6405	5500	0.141	1
6	651	400	0.385	0
7	9135	7200	0.211	1
8	1435	900	0.372	0
9	5922	5500	0.071	1
10	847	1200	0.416	0
11	8050	9600	0.192	1
12	4620	5700	0.233	1
13	2352	2900	0.232	1
14	2174	3000	0.379	0
15	19894	15300	0.230	1
MMRE = 0.256				
PRED(25) = 60%				
MAE = 1049				

2) EXPERIMENT II

The experts performed the estimation of the other 15 selected projects out of 81 by using the proposed BBSEE. where 66 remaining projects were used as historical data, the estimation and the performance evaluation results can be viewed in Table 12.

The results of experiment II imply that the performance of BBSEE is satisfactory in terms of MMRE= 0.17, MAE= 596, and PRED(25) =80%. The lowest value of MMRE (i.e. 0.17<=0.25) and highest value of PRED(25) (80%>=75%) indicate the efficacy of BBSEE [34].

3) COMPARISON

The difference between the actual and estimated effort for experiment 1 by using wide-band Delphi is shown in Figure 4. The difference between the actual and estimated effort of

TABLE 12. BBSEE performance.

Project	Actual	Estimated	MRE	PRED(25)
1	5152	5500	0.067	1
2	5635	5300	0.059	1
3	805	700	0.130	1
4	3829	3100	0.190	1
5	2149	1500	0.302	0
6	2821	3600	0.276	0
7	2569	3000	0.167	1
8	3913	4500	0.150	1
9	7854	8500	0.082	1
10	2422	3000	0.239	1
11	4067	4500	0.106	1
12	9051	9000	0.005	1
13	2282	2500	0.095	1
14	4172	6200	0.486	0
15	4977	6000	0.205	1

MMRE = 0.170
PRED(25) = 80%
MAE = 596

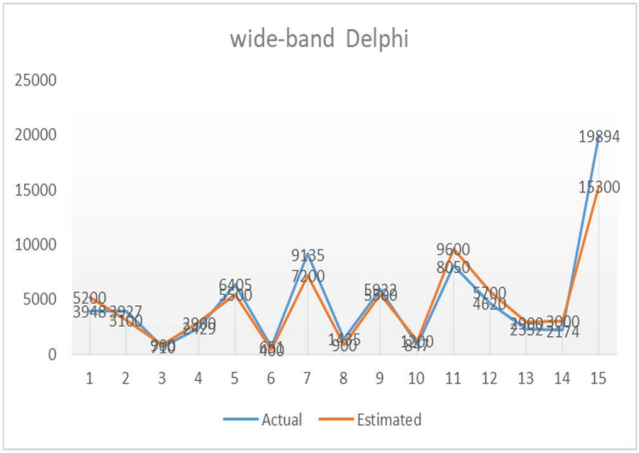

FIGURE 4. Wide-band Delphi predictions.

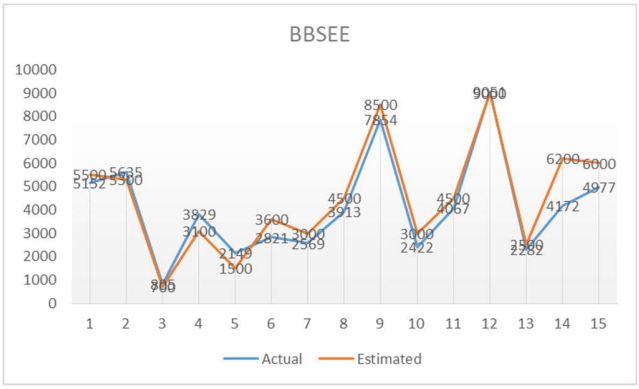

FIGURE 5. BBSEE predictions.

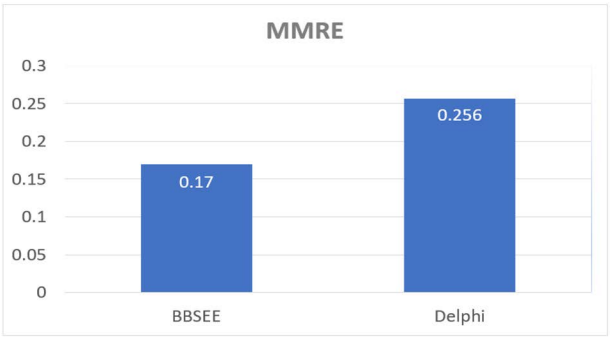

FIGURE 6. Mean magnitude of relative error.

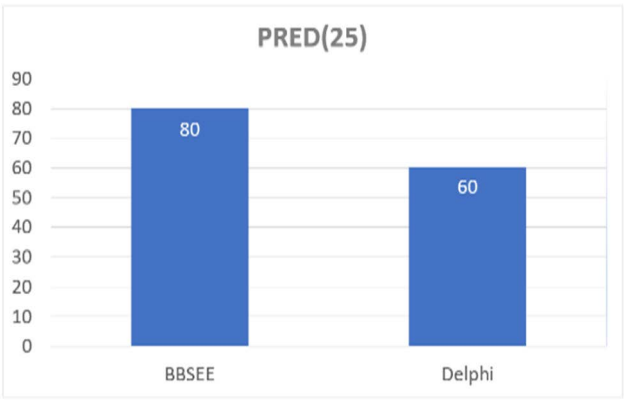

FIGURE 7. Percentage of successful predictions falling.

experiment 2, where the estimation is done by using the BBSEE method is shown in Figure 5.

The main consequences of the results of Figure 4 and Figure 5 are as follows.

- The difference between actual and estimated values of BBSEE is lower than the difference estimated through wide-band Delphi. We observed the lowest value of MMRE (i.e., 0.17) for BBSEE while we observe the value of MMRE as 0.25 for wide-band Delphi. The detailed result is shown in Figure 6.
- We observed the highest value of PRED(25) (i.e. 0.80) for BBSEE while we observe the value of MMRE as 60^ for wide-band Delphi. The detailed result is shown in Figure 7.
- We observed the highest value of MAE (i.e., 1049) for BBSEE while we observe the value of MMRE as 596^ for wide-band Delphi. The detailed result is shown in Figure 8.

4) HOW BBSEE OUTPERFORM

The Delphi technique has some limitations which negatively affect the estimated results such as the inclusion of expert opinion during the estimation process. This limitation is overcome by using the proposed BBSEE method, where the availability of domain experts is certain using the blockchain, and by introducing a freelance effort estimation market where experts are always available. Similarly, the biasness in a physically existing group also negatively affects the results of the Delphi technique. The problem of biasness in the group is overcome by BBSEE by introducing remote groups. Finally, the problem of lack of historical

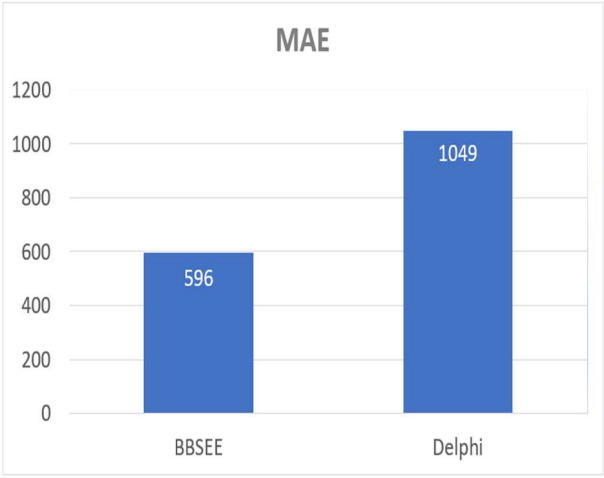

FIGURE 8. Mean absolute error.

data in analogy-based estimation is overcome by BBSEE by introducing a blockchain-based repository, where historical data of all the previous estimations is maintained on the chain for future use. The combined benefits of group and analogy-based estimation improve the estimation results due to which BBSEE outperforms.

5) CONCLUSIVE DISCUSSION

The experimental validation results implied that the proposed BBSEE has overcome the limitation of lack of experts by introducing a freelance estimation market concept, where relevant experts are always available. Similarly, the limitations such as biasness in a group and lack of historical data have been overcome by maintaining a repository on the chain. Finally, the results also imply that the BBSEE has a positive impact on the time and cost required for estimation. The performance evaluation results implied that the proposed BBSEE outperformed as compared to the Delphi technique.

VII. THREATS TO VALIDITY

Empirical approaches are practiced commonly in software engineering, but there is no agreement on how to practice them properly [35]. In this research, the performance of the proposed BBSEE method may vary due to different factors, as the results of this study were dependent on the participating organizations, the estimation experts, evaluation measures, and the selected dataset, some of the threats are described below.

The experts and facilitator perform a very crucial role in the proposed BBSEE Similarly, its performance may vary depending on the level of experience of the experts and facilitator. It also depends on the number of experts involved in the estimation process. The applied dataset also performs a very central role in the performance assessment of the proposed system. The results may also vary with the change of dataset or change in the size of the dataset. The participating organizations also have a significant impact on the results of this research, which is another threat to our study. As we have involved 52 organizations in the case studies, the presented results may vary with an increase or decrease in the number of participants or an increase or decrease in the size of participating organizations. The performance measures used to perform a very central role in estimation results. Therefore, by changing the performance measure the efficiency of the proposed system may vary.

The most difficult decision was about the validation of the proposed model for which there was no better choice than developing a working model of the proposed method. The second difficult decision is of validating the proposed method, so we have selected those organizations which are considered as actively working software industries at the time when the proposed study is conducted.

VIII. CONCLUSION AND FUTURE WORK

The software development effort estimation is one of the crucial software management activities. Due to the importance of software effort estimation, a lot of methods and models are proposed, but there is no best effort estimation method or model is reported. The commonly used estimation methods are classified into a group and analogy-based estimation approaches. However, both have some shortcomings such as historical data, lack of experts, and biasness in a group, which could affect the estimated results. Currently, the research communities are appreciating the use of advanced technologies to improve the estimation results. Due to the emerging situation, this research introduces a Blockchain-Based Software Effort Estimation (BBSEE) method to overcome the shortcomings of the group and analogy-based estimation, and ultimately improve the estimation results.

We have simulated the proposed BBSEE and conducted two case studies using the experts of 52 organizations. The results of the first case study support our assumption that lack of historical data, lack of experts, and biasness in a group are the major barriers and prevent the organizations to perform estimation activities more effectively. The results of the second case study support our assumption that the use of BBSEE could help the project manager overcomes the reported problems such as lack of expertise, biasness in the group members, lack of historical data in the group, and analogy-based estimation. The findings also support the assumption that the proposed BBSEE performs estimation activities more efficiently as compared to traditional methods. The findings of the study also reported that most of the participants voted in favor. Moreover, the proposed BBSEE will decrease the time and cost required for estimation activities.

The Performance of the proposed BBSEE is evaluated by using the Mean Magnitude of Relative Error (MMRE), Mean Absolute Error (MAE), and percentage of successful predictions falling (PRED (25)), we have performed two experiments using the same six experts and using a dataset of 81 projects. In the first experiment, we selected 15 projects and asked experts to use the wide-band Delphi technique to estimate and evaluated the performance. The experimental results in terms of MMRE, MAE, and PRED(25) indicate the efficacy of BBSEE as compared to wide-band Delphi.

In future work, we will enhance our proposed BBSEE methodology by integrating ML-based effort estimation approaches. These future enhancements will run ML-based effort estimation in parallel with the proposed BBSEE method, and the results of ML-based estimation will be used by the experts while making the decisions. Future enhancements are expected to improve the estimation results.

ACKNOWLEDGMENT

The authors are thankful to European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie (No. 801522), the Science Foundation Ireland, and the European Regional Development Fund through the ADAPT Centre for Digital Content Technology (No. 13/RC/2106_P2).

REFERENCES

[1] M. Jorgensen, "Practical guidelines for expert-judgment-based software effort estimation," *IEEE Softw.*, vol. 22, no. 3, pp. 57–63, May/Jun. 2005.
[2] D. Yang, Q. Wang, M. Li, Y. Yang, K. Ye, and J. Du, "A survey on software cost estimation in the Chinese software industry," in *Proc. 2nd ACM-IEEE Int. Symp. Empirical Softw. Eng. Meas. (ESEM)*, Oct. 2008, pp. 253–262.
[3] P. Pospieszny, "Software estimation: Towards prescriptive analytics," in *Proc. 27th Int. Workshop Softw. Meas. 12th Int. Conf. Softw. Process Product Meas.*, Oct. 2017, pp. 221–226.
[4] L. Lavazza and S. Morasca, "On the evaluation of effort estimation models," in *Proc. 21st Int. Conf. Eval. Assessment Softw. Eng.*, Karlskrona, Sweden, Jun. 2017, pp. 41–50.
[5] B. Tanveer, L. Guzmán, and U. M. Engel, "Understanding and improving effort estimation in agile software development: An industrial case study," in *Proc. Int. Conf. Softw. Syst. Process*, May 2016, pp. 41–50.
[6] S. G. International, "The chaos report," Standish Group, Boston, MA, USA, Tech. Rep., 2015. [Online]. Available: https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf
[7] M. Usman, E. Mendes, and J. Börstler, "Effort estimation in agile software development: A survey on the state of the practice," in *Proc. 19th Int. Conf. Eval. Assessment Softw. Eng.*, Nanjing, China, Apr. 2015, pp. 1–10.
[8] S. Munialo and G. Muketha, "A review of agile software effort estimation methods," *Int. J. Comput. Appl. Technol. Res.*, vol. 5, no. 9, pp. 612–618, 2016.
[9] M. Jorgensen, "What we do and don't know about software development effort estimation," *IEEE Softw.*, vol. 31, no. 2, pp. 37–40, Mar. 2014.
[10] S. Shukla, S. Kumar, and P. R. Bal, "Analyzing effect of ensemble models on multi-layer perceptron network for software effort estimation," in *Proc. IEEE World Congr. Services (SERVICES)*, Jul. 2019, pp. 386–387.
[11] B. A. Tama, B. J. Kweka, Y. Park, and K.-H. Rhee, "A critical review of blockchain and its current applications," in *Proc. Int. Conf. Electr. Eng. Comput. Sci. (ICECOS)*, Aug. 2017, pp. 109–113.
[12] R. Chatterjee and R. Chatterjee, "An overview of the emerging technology: Blockchain," in *Proc. 3rd Int. Conf. Comput. Intell. Netw. (CINE)*, Oct. 2017, pp. 126–127.
[13] S. Porru, A. Pinna, M. Marchesi, and R. Tonelli, "Blockchain-oriented software engineering: Challenges and new directions," in *Proc. IEEE/ACM 39th Int. Conf. Softw. Eng. Companion (ICSE-C)*, May 2017, pp. 169–171.
[14] M. Beller and J. Hejderup, "Blockchain-based software engineering," in *Proc. IEEE/ACM 41st Int. Conf. Softw. Engineering: New Ideas Emerg. Results (ICSE-NIER)*, Montreal, QC, Canada, May 2019, pp. 53–56.
[15] R. Saljoughinejad and V. Khatibi, "A new optimized hybrid model based on COCOMO to increase the accuracy of software cost estimation," *J. Adv. Comput. Eng. Technol.*, vol. 4, no. 1, pp. 27–40, 2018.
[16] S. Shekhar and U. Kumar, "Review of various software cost estimation techniques," *Int. J. Comput. Appl.*, vol. 141, no. 11, pp. 31–34, May 2016.
[17] A. Baghel, M. Rathod, and P. Singh, "Software effort estimation using parameter tuned models," 2020, *arXiv:2009.01660*.
[18] J. Huang, Y.-F. Li, J. W. Keung, Y. T. Yu, and W. K. Chan, "An empirical analysis of three-stage data-preprocessing for analogy-based software effort estimation on the ISBSG data," in *Proc. IEEE Int. Conf. Softw. Qual., Rel. Secur. (QRS)*, Jul. 2017, pp. 442–449.
[19] S. Shekhar and U. Kumar, "Review of various software cost estimation techniques," *Int. J. Comput. Appl.*, vol. 141, no. 11, pp. 31–34, May 2016.
[20] P. Serrador and J. K. Pinto, "Does agile work?—A quantitative analysis of agile project success," *Int. J. Project Manag.*, vol. 33, no. 5, pp. 1040–1051, Jul. 2015.
[21] M. Adnan, M. Afzal, and K. H. Asif, "Ontology-oriented software effort estimation system for E-commerce applications based on extreme programming and scrum methodologies," *Comput. J.*, vol. 62, no. 11, 2019, Art. no. 16051624.
[22] M. Adnan and M. Afzal, "Ontology based multiagent effort estimation system for scrum agile method," *IEEE Access*, vol. 5, pp. 25993–26005, 2017.
[23] Z. Polkowski, J. Vora, S. Tanwar, S. Tyagi, P. K. Singh, and Y. Singh, "Machine learning-based software effort estimation: An analysis," in *Proc. 11th Int. Conf. Electron., Comput. Artif. Intell. (ECAI)*, Jun. 2019, pp. 1–6.
[24] J. Wen, S. Li, Z. Lin, Y. Hu, and C. Huang, "Systematic literature review of machine learning based software development effort estimation models," *Inf. Softw. Technol.*, vol. 54, no. 1, pp. 41–59, 2012.
[25] A. B. Nassif, M. Azzeh, L. F. Capretz, and D. Ho, "Neural network models for software development effort estimation: A comparative study," *Neural Comput. Appl.*, vol. 27, no. 8, pp. 2369–2381, Nov. 2016.
[26] V. S. Dave and K. Dutta, "Neural network-based models for software effort estimation: A review," *Artif. Intell. Rev.*, vol. 42, no. 2, pp. 295–307, 2014.
[27] M. Jørgensen, "Forecasting of software development work effort: Evidence on expert judgement and formal models," *Int. J. Forecasting*, vol. 23, no. 3, pp. 449–462, Jul. 2007.
[28] K. Moløkken and M. Jørgensen, "Software effort estimation unstructured group discussion as a method to reduce individual basis," in *Proc. PPIG*, 2003, p. 4.
[29] F. Walkerden and R. Jeffery, "An empirical study of analogy based software effort estimation," Empirical software engineering," *Empirical Softw. Eng.*, vol. 4, no. 2, pp. 135–158, 1999.
[30] Z. Abdelali, H. Mustapha, and N. Abdelwahed, "Investigating the use of random forest in software effort estimation," *Proc. Comput. Sci.*, vol. 148, pp. 343–352, Jan. 2019.
[31] M. Pandey, R. Litoriya, and P. Pandey, "Validation of existing software effort estimation techniques in context with mobile software applications," *Wireless Pers. Commun.*, vol. 110, no. 4, pp. 1659–1677, Feb. 2020.
[32] M. Hosni, A. Idri, and A. Abran, "Evaluating filter fuzzy analogy homogenous ensembles for software development effort estimation," *J. Softw., Evol. Process*, vol. 31, no. 2, p. e2117, Feb. 2019.
[33] J. S. Shirabad and T. J. Menzies, "The PROMISE repository of software engineering databases," School Inf. Technol. Eng., Univ. Ottawa, Ottawa, ON, Canada, 2005. [Online]. Available: http://promise.site.uottawa.ca/SERepository
[34] S. D. Conte, H. E. Dunsmore, and Y. Shen, *Software Engineering Metrics and Models*. San Francisco, CA, USA: Benjamin-Cummings, 1986.
[35] J. Siegmund, N. Siegmund, and S. Apel, "Views on internal and external validity in empirical software engineering," in *Proc. IEEE/ACM 37th IEEE Int. Conf. Softw. Eng.*, May 2015, pp. 9–19.
[36] U. Farooq, M. Ahmed, S. Hussain, F. Hussain, A. Naseem, and K. Aslam, "Blockchain-based software process improvement (BBSPI): An approach for SMEs to perform process improvement," *IEEE Access*, vol. 9, pp. 10426–10442, 2021.

MANSOOR AHMED received the M.Sc. degree in information management from the Vienna University of Technology (TU Wien), Austria, and the Ph.D. degree from Technical University Vienna, Austria, in 2009. In 2010 and 2011, he was awarded a Senior Researcher Fellowship Scholarship by Indiana University, the USA for his postdoctoral studies. In 2017, he worked as a Senior Researcher at University College Dublin (UCD). He is a member of the editorial review board of many international journals. Besides this, he has organized several workshops and also serves as a technical program committee member in international reputed conferences. His research interests include data provenance, blockchain, semantic web technologies, knowledge-based systems, privacy issues in health care data, cloud computing security, information security, and privacy. Currently, he is a Research Fellow of the Innovative Value Institute, Maynooth University, Ireland, under a Marie Sklodowska-Curie Actions (MSCA)/EU research-funded project. He was awarded Higher Education Commission (HEC) Scholarship for Higher Studies (Ph.D.) in Austria.

NAEEM IQBAL (Member, IEEE) received the B.S. degree in computer science from COMSATS University Islamabad, Attock Campus, Punjab, Pakistan, and the M.S. degree in computer science from COMSATS University Islamabad, Attock Campus, in 2019. He is currently pursuing the Ph.D. degree with the Department of Computer Engineering, Jeju National University, Republic of Korea. He has professional experience in the software development industry and academics as well. He has published more than 25 papers in peer-reviewed international journals and conferences. He is serving as a professional reviewer for various well-reputed journals and conferences. His research interests include AI-based intelligent systems, data science, big data analytics, machine learning, deep learning, analysis of optimization algorithms, the IoT, and blockchain-based secured applications.

FARAZ HUSSAIN received the B.S. degree in software engineering from International Islamic University, Islamabad, Pakistan, in 2017. He is currently pursuing the M.S. degree in software engineering with COMSATS University, Islamabad. His research interests include solving trending software engineering problems using advanced technologies, such as blockchain and machine learning.

MURAD-ALI KHAN received the B.S. degree in computer science from COMSATS University Islamabad, Attock Campus, Punjab, Pakistan, in 2020. He is currently pursuing the integrated Ph.D. degree with the Department of Computer Engineering, Jeju National University, Republic of Korea. He has professional experience in the software development industry. His research interests include ML and data mining related applications.

MARKUS HELFERT is currently the Director of the Innovation Value Institute Research Center and a Professor in digital service innovation at Maynooth University. His research interests include PayTech, information management, innovation, cloud computing, service science, enterprise architecture and smart cities, and smart buildings. He collaborates with a range of enterprises in relation to innovation and IT management. He is also the Project Coordinator of the H2020 Projects: PERFORM on Digital Retail and is the Chair of Steering Committee Elite-S H2020 Marie Curie Project on 'Future Leaders in ICT Standardization for Europe. He has authored more than 200 academic articles, journals, and book contributions, and has presented his work at international conferences. He is a member of the International Program Committees and a Reviewer and an Associate Editor of *IS Journals*.

IMRAN received the Ph.D. degree from the Computer Engineering Department, Jeju National University, Republic of Korea. He worked as a Researcher with the MCL and JNU Big Data Laboratory. Currently, he is an Assistant Professor at the Department of Biomedical Engineering, IT Convergence College, Gachon University, Republic of Korea. His research interests include software development, IT convergence-based solutions, entrepreneurship, interdisciplinary scientific applications based on the Internet of Things, machine learning, data science, and blockchain.

JUNGSUK KIM (Member, IEEE) received the B.S. degree *(magna cum laude)* in electrical engineering from Sogang University, Seoul, South Korea, in 2003, the M.S. degree in electrical engineering from the University of Southern California, Los Angeles, in 2006, and the Ph.D. degree from the University of California at Santa Cruz, Santa Cruz, in 2011. After graduating in 2003, he worked at the DRAM Design Laboratory, Samsung Electronics, as an Engineer. He is currently an Associate Professor with the Department of Biomedical Engineering, Gachon University, South Korea. His research interests include integrated circuits and system designs for nanopore sequencing technology and other biomedical applications.