IEEE Access
The journal for rapid open access publishing

Received March 21, 2015, accepted April 15, 2015, date of publication April 20, 2015, date of current version April 30, 2015.

Digital Object Identifier 10.1109/ACCESS.2015.2424703

Computerized Argument Delphi Technique

SADI EVREN SEKER, (Member, IEEE)
Department of Business, Istanbul Medeniyet University, Istanbul 34000, Turkey (academic@sadievrenseker.com)

ABSTRACT The aim of this study is the computerization of the argument Delphi method. The Delphi method is mainly designed for qualitative prediction within a group of experts, where the experts make predictions and a facilitator controls these predictions until the experts end up with a level of consensus. Argument Delphi, as opposed to the classical Delphi model, is built on the contradictions of the ideas of the experts. Argument Delphi mainly focuses on a discussion topic and asks experts to create new arguments and criticize other arguments from other experts. After a certain level of contradiction, the method yields an amount of contradictory, criticized arguments and builds a decision over these antitheses, as in the Hegelian approach. This is the first time the argument Delphi method has been modeled in a graph of arguments and the problem of qualitative decision has been transferred into a graph problem using Delphi method. This paper is also the first time that argument aggregation and evaluation methods have been proposed. Moreover, the computerized version of argument Delphi is applied to real-world problems using crowd involvement through Facebook. The problem is defined as the prediction of petroleum prices for the end of year and more than 100 contributors from all around the world argued and criticized each other. This paper also discusses the findings of this case study.

INDEX TERMS Delphi technique, decision support systems, qualitative methods, graph theory, social network, crowd opinion.

I. INTRODUCTION

The Delphi method is a qualitative prediction of forecasting method, where experts share their expertise on the prediction in a turn-based system [1].

The Delphi method has three major phases as demonstrated in Figure 1. In the first phase, the experts make predictions about the question posed by the facilitator. In the second phase, the facilitator collects and aggregates the predictions. Finally, in the third phase the facilitator announces the aggregated prediction back to the experts and all the experts update their predictions and make new ones. The method runs until a satisfactory level of prediction is reached by repeating all three steps [2].

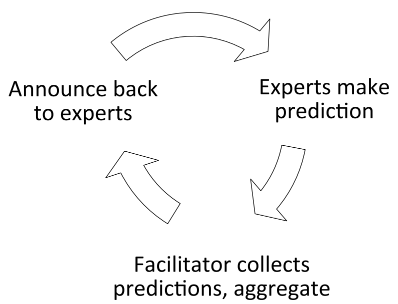

FIGURE 1. Three rounds of Delphi Method.

The Delphi method was first introduced into the literature in 1944 during the World War II to create reports about future military technological capabilities by the US Air Corps [3].

Following its introduction, the Delphi method had a great long-term impact on the literature and it is still very active for many applications all around the world. Most management textbooks and popular management books even cover the method [4]–[7]. Because of its importance and impact on the fields of management and administration for many years, many varieties of the method have been developed and successfully utilized by researchers worldwide. In his study, Yousuf [8] divides the development of the Delphi method into 5 stages. In his division, he has underlined the third stage as taking place between late 1960s to mid 1970s with 389 articles, papers and reports, while Rieger [9] indicates 61 dissertations between 1970 and 1974. According to the divisions proposed by Yousuf [8], in the fifth stage, the number of dissertations on the Delphi technique between 1980 and 1984 was 441. The increasing impact of technique since it was first introduced into the literature can be understood from these statistics.

During the period of this study, according to the Google Scholar statistics, the number of articles reports or dissertations related to the Delphi method just between 2010 and 2014 was more than 22,000.

There is also some recent work by the IS/IT researchers. In one of these researches, researchers examined the impact of the Delphi method from 2502 articles and showed how the Delphi method is useful in different situations [10]. Another recent study involves the forecasting of financial markets by the Delphi method, where the method has been applied in two phases. The first phase corrects the belief perseverance bias, while the second phase tests a conditional forecast process, which unexpectedly proves unsuccessful [11].

Because of the limitations of the traditional Delphi method, there are several varieties which have been introduced into the literature. One famous application—which is conceptually not far from argument Delphi—is policy Delphi [12]. Policy Delphi focuses on generating the strongest possible opposing viewpoints about any policy. Different than the consensus of traditional Delphi, policy Delphi is built on the debates [13]. The beauty of the Delphi method can also be connected to its Hegelian nature [14]. After the first round of the Delphi method, the experts are divided into two camps. For example, consider the weather forecast scenario. Experts are asked for the weather forecast for the next season. After the first round of predictions, the experts are divided into two groups, where the first group has an expectation lower than the average forecast and second group has an expectation higher than the average forecast. At any point, Delphi produces two contrary groups, which can also be treated as the thesis and anti-thesis for the discussion. The weakness of the classical Delphi method is its lack of arguments and only the numbers or the moderator collects the predictions. At any phase the reasons are out of discussion and each expert tries only to check their predictions with common sense. On the other hand, information exchange between experts can convince some of the experts about the prediction of the opposing camp, but unfortunately the classical Delphi method lacks such an alternative.

In addition, another major impact on the Delphi method is the computerized versions [15]. These versions mainly give the advantage of computer technology, such as communication capabilities, automatic selection or elimination, logging and trace back and so on.

One version of the policy Delphi method, named Argument Delphi, also concentrates on the interchange of arguments between experts. The Argument Delphi method is more specifically used for stating the arguments and hosting discussions about the arguments [16]. For example, in an application of the Argument Delphi method on engineering education for Finland 2015, the method starts by interviewing the experts in the first round. Interviewing each expert takes about 1.5 to 3 hours and in the application they have selected only 21 experts. In the following rounds, the experts are asked to give their opinions about the education system via questionnaire [17].

II. COMPUTERIZING ARGUMENT DELPHI METHOD

In this study, a computerized version of Argument Delphi is proposed which is also suitable for web, social media or mobile applications. Besides the text mining [18] or streaming data mining [19] approaches to social media analysis, computerized argument Delphi is proposed as an alternative to gather structured data in order to understand the opinion of the crowd. The major impact of the computerized argument Delphi method is being able to conduct all the operations in a web environment. For example, the collection of arguments from the experts can be collected by interviews, in written form, in discussion groups or by brainstorming [20], [21]. As a computerized version of the argument Delphi method, data collection via the Internet is proposed. Because of its various alternatives, this computerized version of argument Delphi lets the contributors write their opinions, arguments or assumptions freely on the topic. This version of Delphi has major advantages—for example, all the discussions on social networks or web pages are a form of written debate on the topic, such as the cases on Facebook, where one of the users shares an info-graphic and then hundreds of comments are made in the discussion [22]–[24]. In the computerized argument Delphi method, the first (basis) step is as easy as putting up an argument for the discussion. After the first argument about any opinion, anyone in the same environment (like social media, such as Facebook, Twitter, LinkedIn or other web environments like forums, blogs or portals [25], [26]) can get involved in the discussion within the rules of computerized argument Delphi.

Computerized argument Delphi is not a discrete approach, where each step can only take action after the previous step is completed. Instead, the method has a continuous arguing behavior and all steps can take action after the first step. Also, results can be collected at any time and the most important output of the method, called an argument graph, can be collected at any time after the method is started.

According to its continuous behavior, the computerized argument Delphi method allows any contributor to add new arguments about the discussion as the second step and rate the previous arguments as the third step. These three steps keep running and contributors can access the system at any time or any number of times without limitations.

Computerized argument Delphi method has an output phase, which can be executed simultaneously to the three phases of the contributor involvement step. All these steps are demonstrated in figure 2.

The traditional argument aggregation is the weakest part of the argument Delphi method and the strength of arguments is missed during the aggregation. The facilitator is forced to make an aggregation in order to end up with a single or only a few arguments for the next step. The aggregation step either completely loses the majority of the arguments or weakens the effect of a majority of the arguments on the next step, because the arguments from the first step are aggregated and reduced into a single argument.

The argument aggregation of the traditional argument Delphi method can be explained by set theory. Let A and E be the sets of arguments and experts, such as $A = \{arg1, arg2, arg3 \ldots argn\}$ and $E = \{exp1, exp2, exp3 \ldots expn\}$. The aggregation function f is a surjective

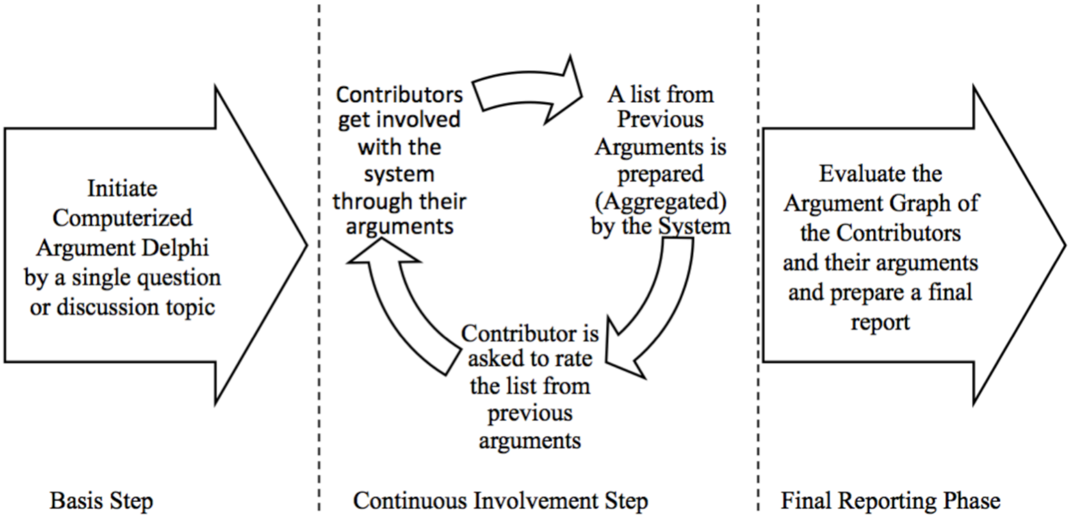

FIGURE 2. Continuous Behaviour of Computerized Argument Map and Three Simultaneous Steps.

function and can be demonstrated by set theory as in equation (1).

$$\forall a \in A, \quad \exists a_{aggregated} \in A, \quad f_{i=1}^{n}(a_i) = a_{aggregated}. \quad (1)$$

In other words, the function f is a one-way function where the aggregated argument can be created from the set of arguments, but the set of arguments cannot be created back from the aggregated arguments. Also, the set member operator in equation (1) is a conceptual approach and it can be considered as an exact set member operator only when one member of set A is selected. The surjective function has only one exceptional case for a two-way possibility, which is the set union on all members of argument set A as demonstrated in equation (2).

$$a_{aggregated} = \bigcup_{i=1}^{n} a_i \quad (2)$$

In the computerized version of the argument Delphi method, it is possible to ask for the opinion of experts about the arguments produced by other experts. In my proposition, the argument generated by one expert is displayed to another expert, and this expert's opinion on the argument is asked by a rating system. After entering their opinion, each expert is asked to rate three random arguments which have been previously entered into the system by other experts.

The proposed approach also can be modeled by set theory. Let A and Rt be two sets of arguments and ratings, respectively, such as $A = \{arg1, arg2, arg3 \ldots argn\}$ and Rt = {r1, r2, r3 …r3n}, where the cardinality of rating set R is three times the cardinality of the argument set A, because for each argument, the contributor rates three previous arguments. Any contribution is a relation between set A and A itself, where the relation has an insertion to the set of arguments A and a rating tuple on set Rt. The contribution relation R can be demonstrated as in equation (3).

$$R : A \rightarrow A$$
$$a, r_1, r_2, r_3 \in A \quad and \quad R \quad (a, r_1, r_2, r_3) \in Rt \quad (3)$$

I have limited the number of the rating to 3 because the computerized version is designed for massive involvement environments like social networks or web pages. For example, it is also possible to ask a contributor to rate all of the previous arguments, but in this case contributor would either spend an excessive amount of time doing so or would prefer to rate the contributions without concentrating on the actual arguments. The number of arguments in further studies can be parametric and the facilitator can decide on the relevant number.

The computerized version of the argument Delphi method can be reviewed from the Hegelian point of view by dividing the contributors into two camps of thesis and anti-thesis [27]. Figure 3 demonstrates the role of each group in the argument Delphi method. After the first prediction,

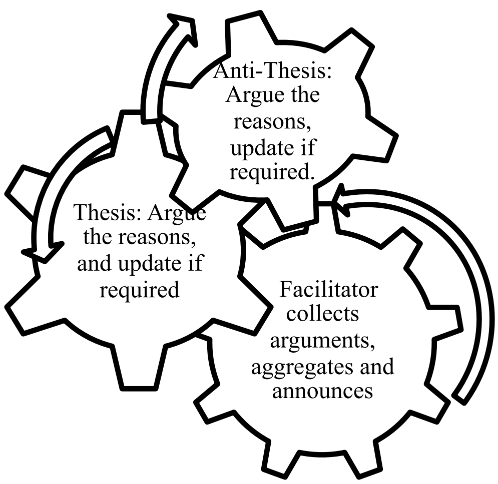

FIGURE 3. Hegelian Approach to Delphi Method.

the experts divide into two camps, the thesis and antithesis camps. Each camp argues the prediction from their point of view, considers the contrary arguments and updates their prediction if they feel they have to. Also, the facilitator keeps collecting predictions, aggregating and announcing them back to the experts.

In the computerized argument Delphi method, a similar approach to dialectics in the Hegelian approach is kept. The only difference—and the biggest impact—of the computerized version is adding multiple arguments into the discussion and giving multiple paths for anybody to follow. Even the contributor can add a new path of dialect for himself.

In figure 4, one of the many possible paths about arguments is demonstrated. The power of the computerized argument Delphi method comes from the possibility of keeping and processing thousands of arguments. Any contributor can easily join the system by making decisions on previously added arguments and can add his own argument to the graph as well.

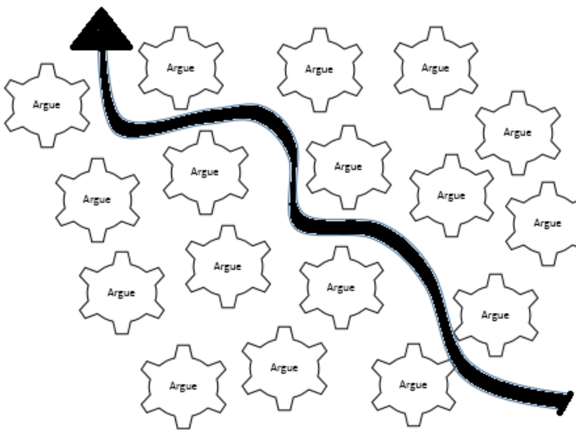

FIGURE 4. Personalized track of argument graph.

III. SAMPLE SCENARIO FOR THE COMPUTERIZED ARGUMENT DELPHI METHOD

The simplicity of the traditional Delphi method is one of its advantages. Everybody can easily understand the Delphi method and can apply it simply to any problem such as prediction, decision support, forecasting, etc. Below is a scenario for the Computerized Argument Delphi Method to help explain it in its most simple form. The scenario is built for a simple prediction of numbers and collects arguments about the prediction. Thus, contributors can argue about their predictions. In addition, after the first step of prediction, contributors can be assumed to be divided into two camps like in the Hegelian approach.

Step 1, Round 1: Facilitator asks question; what is your prediction?

Step 2, Round 1: Contributors write their predictions and send to the facilitator in closed form.

Step 3, Round 2: Facilitator collects the predictions and aggregates them (e.g. average of all predictions until this time) and announces back to the contributors.

Step 4, Round 2: Contributors make another personal prediction after they receive back the aggregated prediction, but the previous aggregation divides the contributors into two camps. The first camp has a lower prediction than the average, while the second camp has a higher prediction than the average. In this case, both contributors in the lower and higher camps may argue about the reasons for their predictions, and why the aggregated prediction should be lower or higher, respectively. So contributors update their predictions by arguing and send them back to the facilitator.

Step 5, Round 3: Facilitator collects the updated predictions and arguments from the contributors. The facilitator then aggregates the predictions (e.g., average) and aggregates the arguments and announces them back to the experts.

Step 6, Round 3: Experts now have the information for the aggregated prediction and arguments from the opposite camp and the supporting camp. They can think about these arguments and, according to their judgment, they can make more sensitive and intelligent predictions. The scenario can also be demonstrated in figure 5.

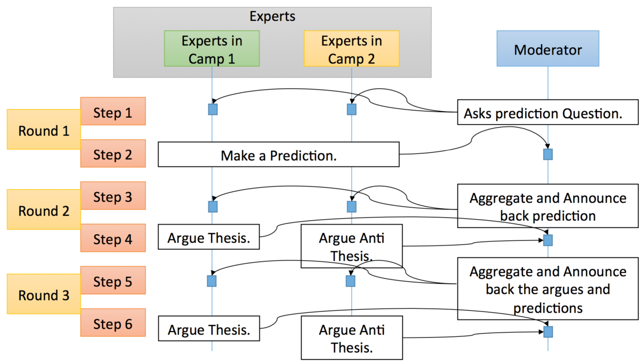

FIGURE 5. Running Scenario of Computerized Argument Delphi, Argument Aggregation.

During the development of the computerized argument Delphi method, one problem we faced was the aggregation of arguments. As a solution, below a list of aggregation methods are proposed.

List of arguments: In this aggregation method, all the arguments are collected and listed together. For example, the facilitator says, people who believe the forecasting is low say "argue1, argue2, argue3 ..." and people who believe the forecasting is high say "argue10, argue11, argue12, ..." etc. This method requires less time for the facilitator but requires a higher amount of time for the panelists in order to read all and evaluate all of the arguments. This can be represented in set theory as in equation (4), similar to equation (2).

$$a_{aggregated} = \bigcup_{i=1}^{n} a_i \tag{4}$$

Drop similar: In this aggregation method, the facilitator reads and eliminates arguments in the same way. For example, if contributor 1 and contributor 2 have similar arguments, the facilitator simply drops one of them, using one of the arguments only. This can be represented in set theory as in equation (5).

$$a_{aggregated} = \bigcup_{i=1}^{n} a_i, \quad if\ any\ D^n_{j=i+1}(a_i,a_j) > t \tag{5}$$

In equation (5), the aggregated arguments are prepared from the set of arguments, where all the arguments are tested through a distance function D and the argument is accepted to the aggregated result only if the distance is above a certain threshold value t, which is defined by the facilitator. The distance function can be based on a natural language processing or a string distance function [28], [29].

Tournament selection: In this aggregation method, one expert, perhaps the facilitator, reads and decides the best arguments as thesis and antithesis, depending on his knowledge and expertise. The aggregation method can also be based on the evolutionary computation [30].

Use qualitative marginal: In this aggregation method, the facilitator uses the arguments from the top n highest and lowest forecasts from the experts. For example, if the n in this method is 2, then the facilitator prepares a list of arguments from the highest two and lowest two forecasting arguments.

$$a_{aggregated} = \bigcup \max_{k} R^n_{\prime=1} a_i \tag{6}$$

The aggregated argument is the set union of k maximum arguments in the rating.

Algorithmic Approach to the Argument Delphi Method.

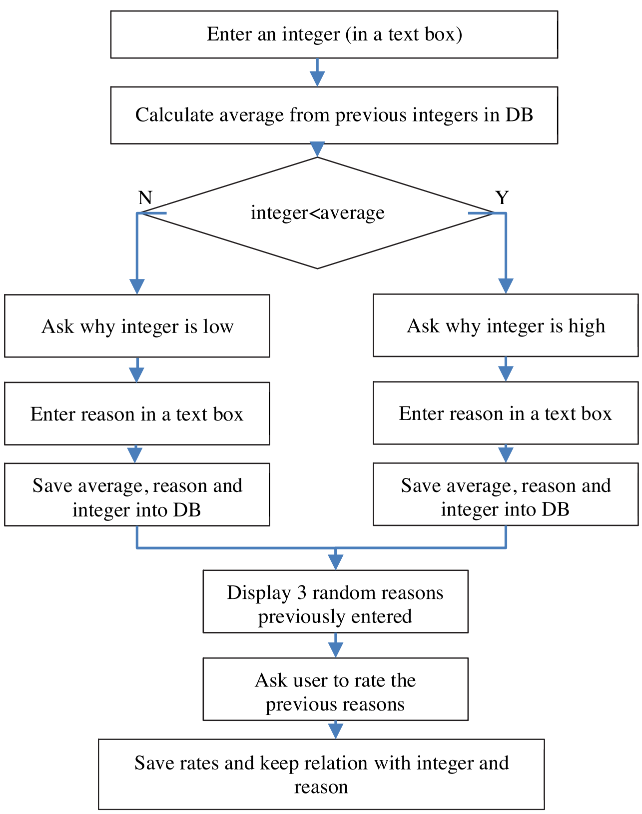

FIGURE 6. Flowchart of a sample computerized argumnet Delphi method.

The computerized argument Delphi method has 3 screens connected to each other and each screen plays a major role in the argument evaluation.

The screen design shown in figure 6 has an algorithm flow chart, which is also demonstrated in figure 7.

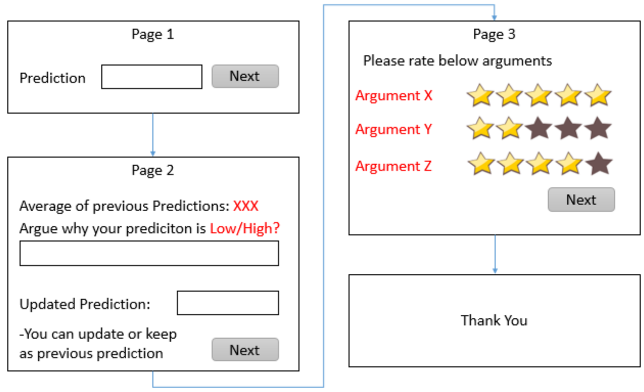

FIGURE 7. Screen Designs from a sample implementation of the computerized argument Delphi method.

The flow chart of the algorithm can also be written as in Algorithm 1.

Algorithm 1 Continuous Contribution Algorithm

Input: Prediction value, as p,
Argument, as a,
Output: Argument a, Prediction p, Ratings set $R=\{ <a_1, r_1> , <a_2, r_2> , <a_3, r_3> \}$
Pre-calculation: n, the number of arguments in database
1: p ← Read prediction from user
2: p_{avg} ← Calculate average ($p_1, \ldots, p_n$)
3: **if** p < p_{avg} **then**
4: a ← Ask argument, why is your prediction low?
5: **else if** p > p_{avg} **then**
6: a ← Ask argument, why is your prediction high?
7: **end if**
8: **for i := 1 to 3**
8: $a_i \leftarrow rand(a_1, \ldots a_n)$
9: r_i ← Ask rating for a_i
10: **end for**
11: **save** record $<a, p, \{<a_1, r_1> , <a_2, r_2> , <a_3, r_3>\}>$

The computerized argument Delphi method gives an opportunity to create a report with arguments. For example, during the prediction period, the average prediction value from the contributors can value the argument and a report can be generated based on the average predictions.

IV. EVALUATION OF ARGUMENT GRAPH

Questions about any argument graph include "Which argument is most valuable and most accepted?", "How can we evaluate all the arguments and summarize them into a single report?", and "What is the most efficient way of aggregating the graphs?".

IEEE Access
The journal for rapid open access publishing

In the study, possible evaluation methods are researched and listed below:

1. Sort and Rate
2. Propagated Path Sort
3. Average Propagation Sort
4. Leave Lowest Out
5. Two Camps Approach

The Sort and Rate approach sorts arguments by their average rating from the other contributors. The argument with highest rating or highest-k ratings can be selected as a result of the computerized argument Delphi method.

Aside from some possible naïve analyses, the questions can be answered via graph theory.

During the evaluation of the computerized argument Delphi method, the arguments and ratings between arguments creates a weighted directed acyclic graph, which is a weighted tree. The argument can be represented via graph theory as $G = (V, E)$, where V is the set of vertices (arguments) and E is the set of edges (ratings). The order of the graph can be defined as $|V|$ and the size of the graph is $|E|$, where $|V|$ is the number of vertices and $|E|$ is the number of edges. The degree of vertices is always three except for the root nodes, so the argument graph can be considered as a 3-regular graph. Although the regularity of the graph is parametric and can be considered as k-regular for the number of arguments rated after each entrance, the sample will be demonstrated over a 3-regular graph. Another assumption about graph G is the weights of the edges. In the proposed graph, the weight of an edge is defined as between 1 and 5 because the argument rating is from 1 to 5 and can be represented with $w \in R \; and \; 1 \leq w \leq 5$.

The evaluation of the argument graph will be explained through a sample argument graph G_1 as represented below:

$$\begin{aligned} G_1 &= (V_1, E_1) \\ V_1 &= \{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8\} \\ E_1 &= \{<A_4, A_1, 5>, <A_4, A_2, 3>, <A_4, A_3, 5>, \\ &\quad <A_5, A_4, 3>, <A_5, A_2, 2>, <A_5, A_3, 5>, \\ &\quad <A_6, A_5, 2>, <A_6, A_2, 3>, <A_6, A_3, 1>, \\ &\quad <A_7, A_4, 5>, <A_7, A_2, 5>, <A_7, A_5, 5>, \\ &\quad <A_8, A_7, 1>, <A_8, A_5, 3.5>, <A_8, A_6, 4.5>\} \end{aligned}$$

Any tuple t in E_1 can be defined as below:

$$t = <e_1 e_2 w>, \; where \; (e_1, e_2) \in E \times E \quad and \; 1 \leq w \leq 5$$

The above graph representation can also be visually demonstrated as in figure 8.

The sort and rate approach for graph G_1 can be calculated by simply calculating the average in-degree of each node in the graph. In-degree can be calculated as demonstrated in equation (7):

$$\deg^{-}(u) = \sum_{u \in V} (u) \tag{7}$$

In the graph theory, "deg−" indicates the in-degree and "deg+" is the out-degree of any node, which is

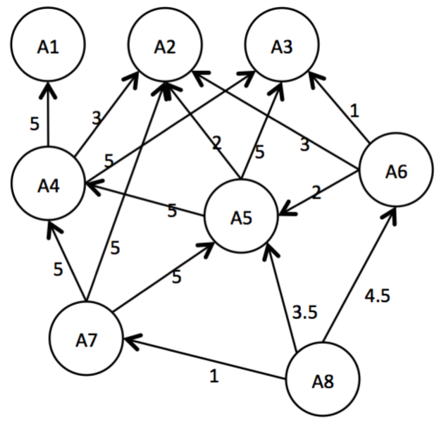

FIGURE 8. Visualization of Sample Argument Graph G1.

an element of the vector set of graph [31]. During this research, there is no interest in the calculation of out-degree. Moreover, the average in-degree is taken into account, and the below equation (8) is proposed for the calculation in the "sort and rate" approach.

$$deg\,(u) = \sum_{u \in V} \frac{w(u)}{|V|} \tag{8}$$

The degree calculation is modified for the weighted graph with weights on the edges and the average is calculated simply by dividing the element count of set V. The calculation for each node is given below:

$$\begin{aligned} &\deg(A_1) = 5, \quad \deg(A_2) = 3.25, \quad \deg(A_3) = 3.66, \\ &\deg(A_4) = 4, \quad \deg(A_5) = 3.5, \quad \deg(A_6) = 4.5, \\ &\deg(A_7) = 1, \quad \deg(A_8) = 0 \end{aligned}$$

from the degree calculation, the highest three arguments in the graph are A_6, A_4, and A_5.

For the directed graph as an argument graph, it is possible to define a directed path, dipath [32] as a sequence of vertices or edges. In this paper the notation of edges will be used to represent the dipaths. In addition, it is possible to define the weight or length of a dipath as in equation (9).

$$w\,(P) = \sum_{,i=0}^{k-1} w(e_i) \tag{9}$$

where P indicates any directed path on the argument graph and w is the weight of any edge $e \in E$. All nodes have a path from root nodes on the $G_{1,}$ which can be calculated as below:

$$\begin{aligned} P_{A1} &= \{P_{A4} \cup <A_4, A_1, 5>\} \\ P_{A2} &= \{P_{A4} \cup <A_4, A_2, 3>, P_{A5} \cup <A_5, A_2, 2>, \\ &\qquad P_{A7} \cup <A_7, A_2, 5>, P_{A6} \cup <A_6, A_2, 3>\} \\ P_{A3} &= \{P_{A6} \cup <A_6, A_3, 1>, P_{A5} \cup <A_5, A_3, 5>, \\ &\qquad P_{A4} \cup <A_4, A_3, 5>\} \\ P_{A4} &= \{P_{A7} \cup <A_7, A_4, 5>\} \\ P_{A5} &= \{<A_8, A_5, 3.5>, <A_6, A_5, 2>, <A_7, A_5, 5>\} \\ P_{A6} &= \{<A_8, A_6, 4.5>\} \\ P_{A7} &= \{<A_8, A_7, 1>\} \\ P_{A8} &= \{\} \end{aligned}$$

Using the above set of paths, the weight of each path can be calculated as below:

$$w(P_{A1}) = 11,\quad w(P_{A2}) = 35,\quad w(P_{A3}) = 32,$$
$$w(P_{A4}) = 6,\quad w(P_{A5}) = 10.5,\quad w(P_{A6}) = 4.5,$$
$$w(P_{A7}) = 1,\quad w(P_{A8}) = 0$$

From the above calculation, the highest path weight is $w(P_{A2})$ with 35. This approach is proposed as **"propagated path sort"** in this paper.

On the other hand, the above classical graph theory approach would be a mistake because the first argument in the argument graph will be the leaf with the highest path weight in most cases, which can also be observed in the sample graph G_1, where the highest three weights are the leaf nodes A_1, A_2 and A_3.

As a solution to the above path length problem, an average weight solution is proposed. For an argument graph, an average weight can be applied in two ways. The first approach can be the mean calculation via equation (10) below:

$$w_{avg}(P) = \sum_{,i=0}^{k-1} \frac{w(e_i)}{k-1} \tag{10}$$

If equation (10) is applied over the weights of all the paths, the values would be as follows:

$$w_{avg}(P_{A1}) = 3.66,\quad w_{avg}(P_{A2}) = 3.33,$$
$$w_{avg}(P_{A3}) = 3.66,\quad w_{avg}(P_{A4}) = 3,$$
$$w_{avg}(P_{A5}) = 3.5,\quad w_{avg}(P_{A6}) = 4.5,$$
$$w_{avg}(P_{A7}) = 1,\quad w_{avg}(P_{A8}) = 0.$$

The average calculation makes the weights normalized between $1 \leq w_{avg} \leq 5$, which makes them comparable.

Another normalization method is based on the qualitative nature of the computerized argument Delphi method, where the weights between nodes are considered as positive and negative. For example, a rating of 1 means the contributor does not share the idea of the argument or else the contributor is against the argument. From this point of view, an average edge weight can be calculated and the path weights can be normalized by the edge weight average. The edge weight average can be calculated as in equation (11):

$$e_{avg}(G) = \sum_{,i=0}^{n} \frac{w(e_i)}{n}, \quad where \quad n = |G| \ and \ e_i \in E \tag{11}$$

During the calculation of e_{avg} the approach is slightly different than the calculation of $w_{avg}(P)$ and the average is calculated via all the edges instead of the edges of the path as in equation (11). After calculating the average weight in graph G, the path weights can be normalized as demonstrated below in formula (12):

$$w_a(P) = \sum_{i=0}^{k-1} w(e_i) - e_{avg} \tag{12}$$

From equation (12), the values of each path can be normalized as below:

$$w_a(P_{A1}) = 0.4,\quad w_a(P_{A2}) = -3.86,$$
$$w_a(P_{A3}) = 0.2,\quad w_a(P_{A4}) = -1.06,$$
$$w_a(P_{A5}) = -0.1,\quad w_a(P_{A6}) = 0.96,$$
$$w_a(P_{A7}) = -2.53,\quad w_a(P_{A8}) = 0.$$

The negative values of path weights can be considered as "disliked" arguments by the society—for example, an argument against the collective opinion of the social network society, like Facebook, Twitter or LinkedIn. From the final results of normalized path weight via average weight, the best three paths can be considered as P_{A6} , P_{A1}, P_{A2} and P_{A3}, and the rest are less than or equal to zero. This approach is proposed with the name of **"average propagation sort"** in this study.

Another solution proposed in this paper is called the **"leave lowest out"** technique. The motivation behind this technique is the dirty data insertion in the social networks, like empty fields, uncompleted forms or erroneous entrances on purpose. Since the argument Delphi method was originally designed for the experts, application of the method over massive user environments like social media or web pages brings such problems. The computerized version brings the solution from the problem itself and uses the contributors to evaluate and understand the dirty data in the system. One major reason of rating an argument to a "1", which is the lowest rating possible, is because it is dirty data and contributors don't like dirty data rating.

The **"leave lowest out"** technique was originally developed for understanding and removing dirty data from the system, but later on the technique evolved to compute the winning arguments by removing all the lowest arguments until the winner is left.

This technique simply starts from the argument with the lowest rating and then deletes it from the system. One important issue during the deletion of any argument is removing all the ratings from the first argument to other arguments. This means that when an argument is removed, all connected arguments are affected and there is a possibility of changing the second lowest argument.

The technique keeps running until the desired top k arguments are left in the system. Some possibilities during the deletion of arguments from the system are demonstrated over the graph visualization.

The degree of each node is already calculated in the "sort and rate" approach as below:

$$\deg(A_1) = 5,\quad \deg(A_2) = 3.25,\quad \deg(A_3) = 3.66,$$
$$\deg(A_4) = 4,\quad \deg(A_5) = 3.5,\quad \deg(A_6) = 4.5,$$
$$\deg(A_7) = 1$$

From the list, the lowest degree is A_7 with 1 and the second lowest is A_2 with 3.25. The removal of the lowest node A_7 is demonstrated in figure 9:

The removal of node A_7 triggers the removal of the edges from this node and also the edges connecting to this node.

IEEE Access
The journal for rapid open access publishing

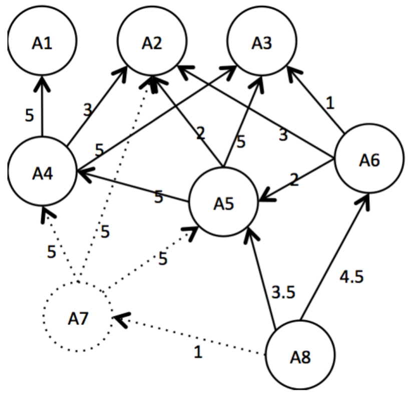

FIGURE 9. Removal of an argument from argument graph.

After the removal, the averages are recalculated and the degrees are as shown below:

$$\deg(A_1) = 1, \quad \deg(A_2) = 3, \quad \deg(A_3) = 3,$$
$$\deg(A_4) = 1, \quad \deg(A_5) = 2, \quad \deg(A_6) = 1$$

Previously the lowest second node was A_2 with 3.25, but after the removal of node A_7 the lowest nodes are A_1, A_4 and A_6. After the removal of these three nodes, the graph will end up with only the A_2, A_3 and A_5 arguments. Using the ''leave lowest out'' technique, those three nodes can be declared the winning arguments.

The final technique proposed in this study is the **''two camps approach''**. In this approach, the arguments are divided into two camps according to their approach to the prediction. One camp has arguments below the average and the other camp has arguments above the average. The ''two camps approach'' divides these arguments into two camps from this respect and assumes the opposite camps will rate these arguments negatively. On the other hand, supporting arguments from the same camp would be expected to be high.

The application of the ''two camps approach'' to the sample graph G_1, requires the calculation of difference from the prediction and the average of predictions. The calculation of prediction differences for each argument is calculated by equation 13:

$$diff(A) = P_{avg} - P_A \tag{13}$$

In order to calculate the differences, the prediction values are given as below in sample graph G_1.

Values on the nodes of figure 10 indicate the prediction values by the contributor of the same argument. The average prediction value of the argument graph in figure 10 is 66.25. The difference for each node can be listed as: $diff(A_1) = -16.25$, $diff(A_2) = 3.75$, $diff(A_3) = 13.75$, $diff(A_4) = -21.25$, $diff(A_5) = 8.75$, $diff(A_6) = -1.25$, $diff(A_7) = -1.25$, $diff(A_8) = 3.75$.

From the list of differences, the arguments can be divided into 2 camps—positive and negative—and the camps can be grouped as below:

Positive camp: A_2, A_3, A_5, A_8

Negative camp: A_1, A_4, A_6, A_7

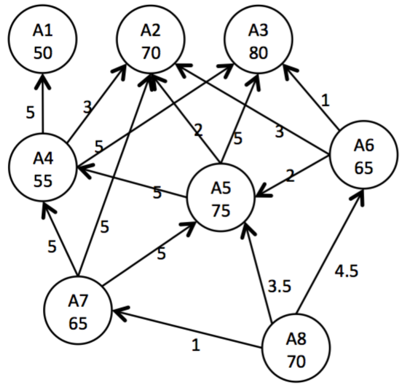

FIGURE 10. Prediction values together with the arguments.

The ratings between opposite camps are signed negatively and the rating values are calculated by the degree equation given in the ''sort and rate'' approach. The updated graph with the signs is also demonstrated in figure 11:

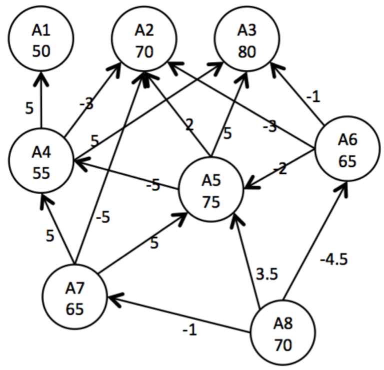

FIGURE 11. Updated weights by the predictions.

Updated degree values are: $\deg(A_1) = 5$, $\deg(A_2) = -3$, $\deg(A_3) = -3$, $\deg(A_4) = 0$, $\deg(A_5) = 2.16$, $\deg(A_6) = -4.5$, $\deg(A_7) = -1$.

From the above list, the highest three arguments are A_1, A_4 and A_5, rather than arguments A_4, A_5 and A_6 as seen in the classical ''sort and rate'' approach.

V. CRITIQUE OF COMPUTERIZED ARGUMENT DELPHI

This section prepares for criticism of the computerized argument Delphi method. Advantages, disadvantages, weaknesses and strengths of the method will be discussed along with the problems and proposed solutions faced during the creation of the method. This section is organized as follows:

1. Properties, strengths and weaknesses of the computerized argument Delphi method
2. First come gets higher attention
3. Dirty data

Besides the above problems, it is possible to apply any graph analysis technique like Page Rank [33], modularity, HITS [34], Eigen Vector Centrality [35], etc. These techniques are already included in the literature and details will not be covered in this paper. However, their applications will be covered in the real problem in section 5.

A. STRENGTHS AND WEAKNESSES OF THE COMPUTERIZED ARGUMENT DELPHI METHOD

The computerized version of argument Delphi can be discussed based on the following aspects:

- Anonymity: In the traditional Delphi method, the identity of the expert is not revealed. The method is useful for environments where identities can be kept secret, like social media or web pages. During the development of the computerized argument Delphi method, because of several problems faced, a few different variations are studied. First, the computerized argument Delphi method has two alternatives from the perspective of anonymity and authentication.
  - Anonymized Identity
  - Authorized Identity

In the anonymized version, the users are not required to login into the system. Each user has only one opportunity to get involved in the system and they simply contribute by adding a new argument about the system and rating the three previous arguments already in the system. Users are not able to update their previous answers, but they can get involved in the system multiple times and affect the final results of the system.

In the authorized identity version, users can access the system through an authorization such as a username/password or other credential like Facebook login, Twitter login, etc. After they get into the system, they are able to contribute, similar to the anonymized version. They are free to rate any number of arguments that have previously been entered into the system. They can come back and update their opinions at any time.

- Structured flow of information: The steps, involvement of moderation, role of experts and limitations are well-defined. This also reduces the negative group effects of face-to-face discussions. From this perspective, the method can be applied only if the information flow, roles and steps to the final output are well defined. In the computerized version of argument Delphi, the contributors can even be from all around the world and they can be far away from face-to-face communication.
- Regular Feedback: Because the identities of the contributors are hidden, the arguments of contributors are isolated from their identity. This isolation has the advantage of allowing contributors to revise their opinions at any time. The Delphi method prevents this negative effect and allows contributors to revise their opinions at any time. This option is also available in the computerized version of the argument Delphi method through two possibilities. In the public version of the computerized Delphi method, where there is no authentication required, users can contribute to the argument graph many times to affect the map multiple times with multiple aspects. On the other hand, the private version of the computerized argument Delphi method allows the moderator to assign authentication for the contributors and allow them to get involved in the system only a limited number of times.
- Role of facilitator: In the traditional Delphi method, the facilitator has three basic roles, which are creation of the initial question, aggregation of predictions/arguments, and deciding when to finish. All three roles also apply for the computerized version of argument Delphi. Furthermore, the computerized version gives the possibility of automatizing the jobs of the facilitator. For example, the aggregation of arguments can be done automatically via one of the aggregation methods proposed in section 2.2 of this paper. In addition, the finishing condition is no longer required in computerized version, since the results can be collected at any time because of its continuous behavior model.

B. FIRST COME GETS HIGHER ATTENTION

One major problem about the computerized argument Delphi method is the aspect of "first come gets higher attention," which means the first arguments in the system will be displayed more than the last-entered arguments if the arguments are truly displayed with a fair random generator. The reason is because each time an expert rates an argument, he will rate the arguments that have previously been entered. Obviously the first-entered argument will have a chance to be displayed for all the ratings; on the other hand a fresh, newly-entered argument will never be rated.

The directed acyclic feature of the argument graph is also one of the major impacts of the computerized argument Delphi method, so the solution proposed is using an unfair random generator on the behalf of newly entered arguments. This solution both keeps the argument graph acyclic and at the same time gives a higher probability to the less-rated arguments.

The random generator algorithm has been modified as a weighted random generator, where the last-entered arguments have a higher possibility to be picked up for rating.

$$Prob\,(A_i) = i \tag{14}$$

In equation (14), i is the index of argument A and prob function returns the probability of the argument to be picked up for the next rating.

C. DIRTY DATA

One major problem during the evaluation of the argument graph is the effect of spam, errors, and unrelated arguments. Because the computerized argument Delphi is mainly designed for crowd sourcing, the possibility of abuse or unrelated entrance is always possible. During the study, the problem has been solved by two approaches. Initially all the arguments are moderated and if any unrelated argument is entered into the system, it is removed by the moderation. Unfortunately, the increased number of participants makes this approach ineffective. Instead, another solution is applied by putting a report button next to the argument ratings. In this way any participant can mark an argument as unrelated and this argument is then directly reported for moderation.

The removal of unrelated arguments also brings other problems. The first problem is arguments without weights since each time an argument is added into the argument graph, the last added argument has no chance to get a weight. Also there is a chance of having other arguments not rated yet, because the last added argument is selecting 3 random arguments from the argument graph and during the selection there is a possibility of not selecting the one most recently added, which means this previously added argument also has no rating.

As a solution, the unrated arguments are accepted at an average weight. Another solution might be getting them as zero but in this case the removal would start from the most recently added node and all the nodes would be removed in the reverse order of addition.

Also, one problem during the removal of the arguments with the lowest ratings is the possibility of an argument with no rating in the middle of the graph. This can happen if all the nodes connected to this argument are removed. Again, one solution for this problem is to consider the average weight of the argument.

While removing the argument, the argument ratings from the same participant should also be deleted. This problem was the motivation behind the ''leave lowest out'' method explained above in section 3.

VI. REAL PROBLEM: PREDICTION OF PETROLEUM PRICES

After the definition of the problem and first time solutions to the computerization of the argument Delphi method, the method is applied to a real-life prediction problem. For the problem, a web page is coded and the link is shared on social networks. Within a few minutes, more than 100 contributors reply with feedback. It takes about 1 minute to 5 minutes for a contributor to get involved with the prediction. After removing the dirty data like unfinished attempts or null entries, only 63 of them are accepted as clean data. The scenario generated for test purposes was arguing about predicting petroleum prices. Computerized argument Delphi has successfully transformed all 63 of these arguments into an argument graph. After the transformation into a graph, it is now possible to apply classical graph metrics to this data.

A. WEB INTERFACE

The web interface of computerized argument Delphi is implemented with PHP/MySQL technologies. The web page has three phases as previously covered in section 2.3, and screen shots are given in figure 12.

Initially, the user is informed of the petroleum prices since 2001 and asked for a prediction for the end of 2014. The user can enter his prediction in the text box at the end of the page as shown in figure 12.

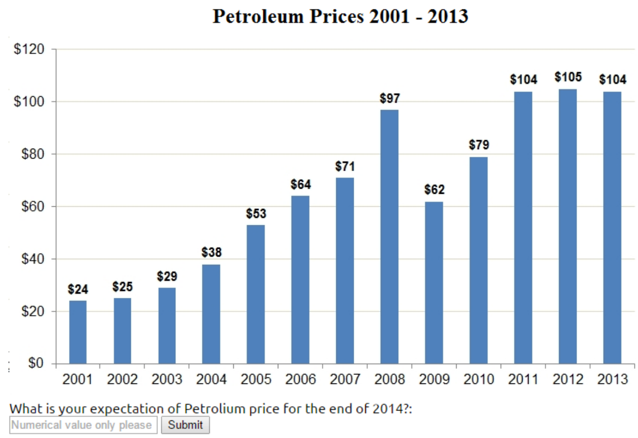

FIGURE 12. First Page of the Web Application.

In the second step, the user is asked to argue about his position. He can either be in the negative or the positive camp, as in figure 13.

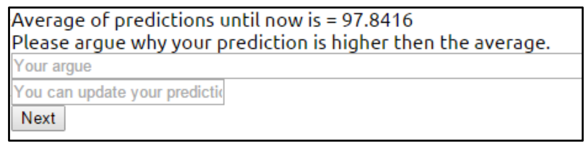

FIGURE 13. Screenshot of the Second Step.

Finally, the contributor is asked to rate three randomly selected previous entries, as in figure 14. Finally all data is stored in the database, which can be reported from the admin interface, as shown in figure 15.

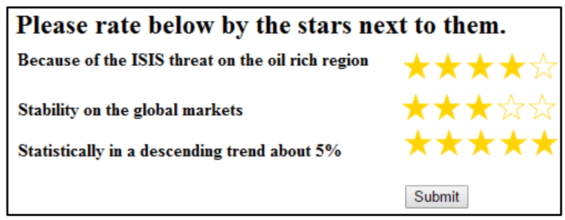

FIGURE 14. Final Rating Step of the Application.

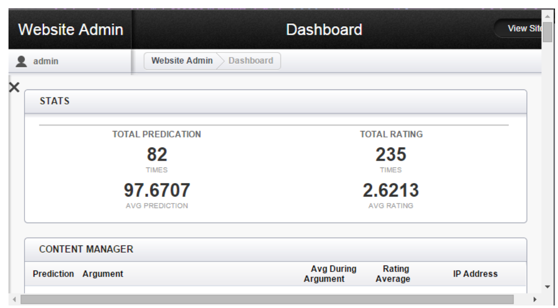

FIGURE 15. Screenshot of the Administrator Interface.

B. GRAPH DEMONSTRATION

The data collected from users are visualized in a graph as shown in figure 16. The graph is weighted and the thickness of the edges indicates the weight of the edge from 1 to 5.

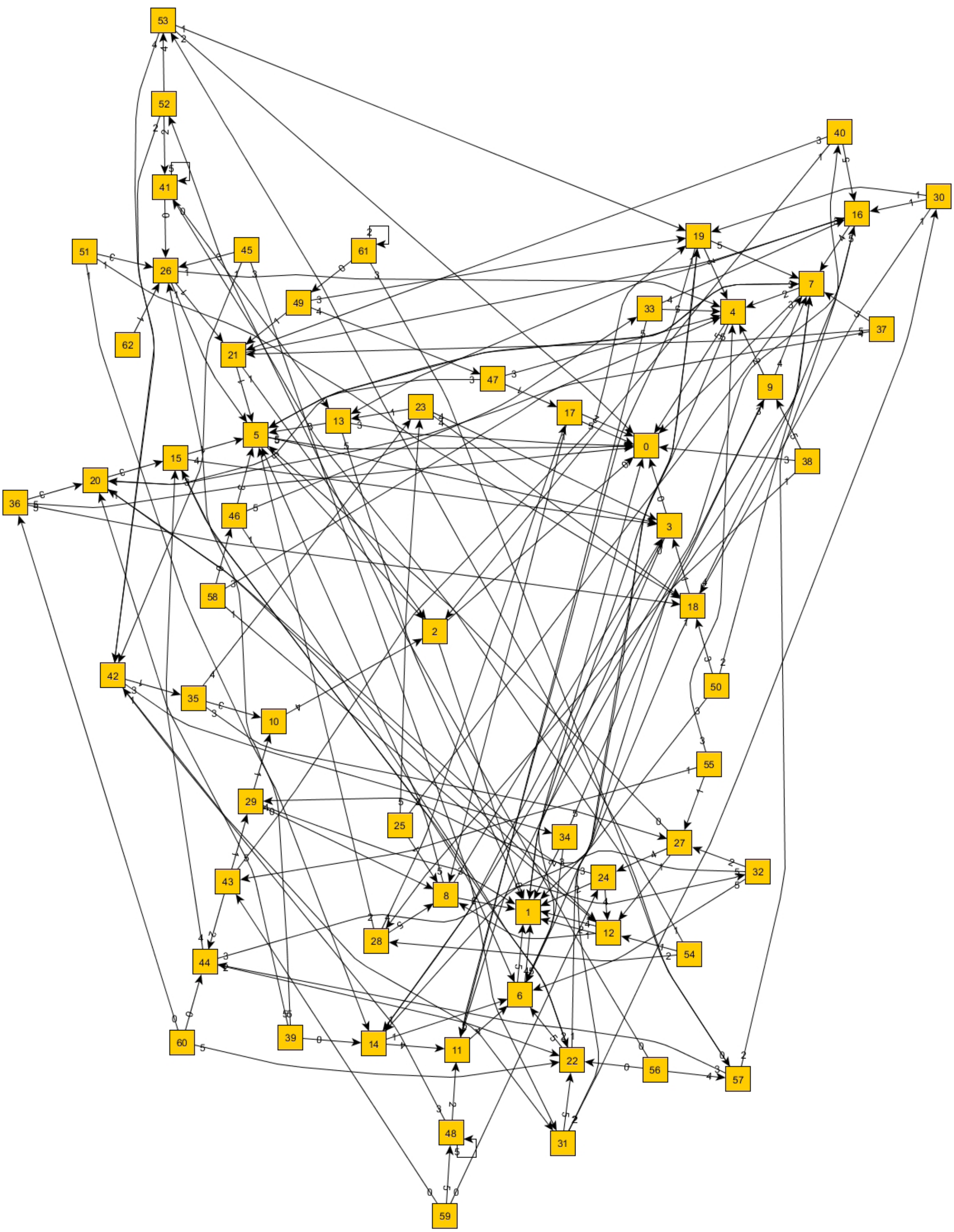

FIGURE 16. Graph Visualization for Petroleum Prediction.

In addition each node is randomly distributed on the graph by the visualization software yED [36].

After converting the arguments from the contributors into a graph, it is possible to make a graph-based analysis. Two groups of approaches will be covered in this paper. The first group of approaches in this study is for understanding the features of the graph and classical graph analysis methods will be applied. The second group of

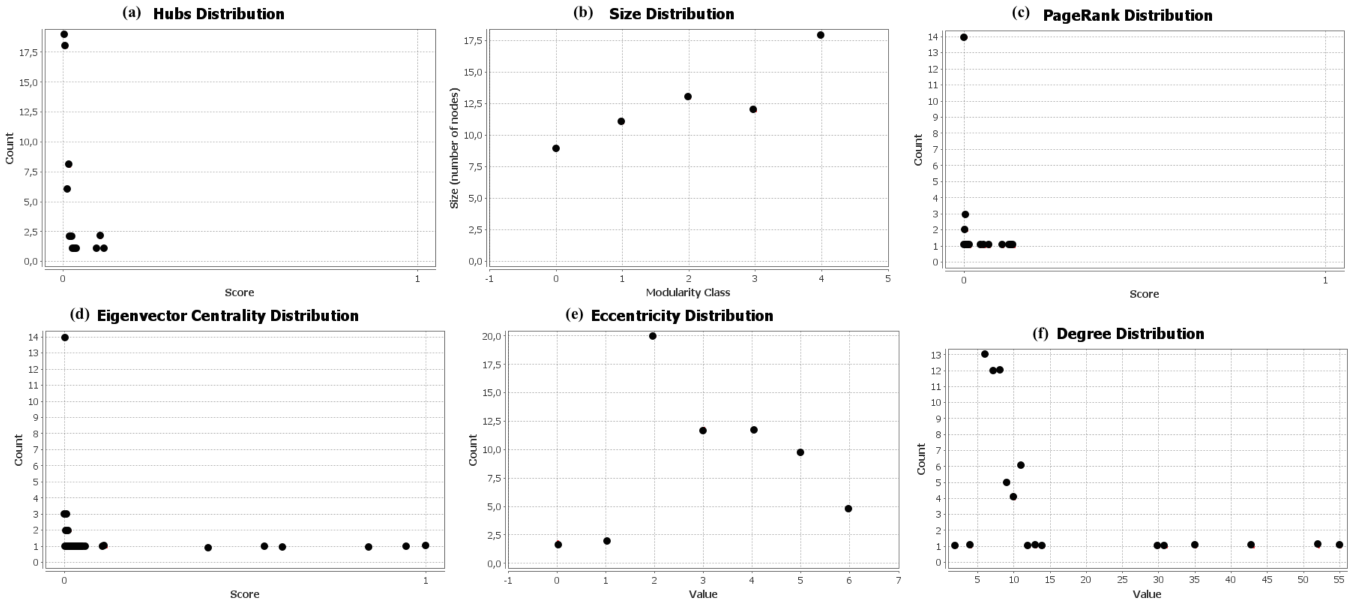

FIGURE 17. Graph Metrics of Case Study.

graph analysis approaches includes the novel approaches proposed for the computerized argument Delphi method in this paper.

C. GRAPH METRICS

After successfully converting all clean argument entries into a graph with the argument ratings, the graph metrics are collected as shown in figure 17.

Metrics of the graph are calculated using Gephi [37] software. Figure 17-(a) indicates the Hubs Distribution, which is also known as hyper-link induced topic search (HITS) [34]. Initially an algorithm is generated to calculate the impact of journal articles or web pages by considering their incoming and outgoing links. For the HUBS distribution, each node starts with a score value of 1 and the number of links are considered with the weights of their incoming links. The algorithm calculates the total score via the outgoing links by considering the incoming links. So the importance of the node is considered to be higher if the score is increased [34]. Similar to the Hubs Distribution, page rank in figure 17-(c) is originally developed for calculating the impact of the web pages by using Google [33]. Again, the higher page rank means a higher impact of a web page or node value in the argument graph. After converting the Delphi arguments into an argument graph, now it is possible to apply a citation impact or web impact algorithm over the algorithms. Arguments also can be queried via these impact algorithms.

Besides the classical graph metrics like degree distribution, degree of nodes [38] as shown in figure 17-(f), Eigen vector Centrality Distribution [35] as shown in figure 17-(d), and eccentricity distribution [39] as shown in figure 17-(e) can easily be applied to the arguments gathered from computerized argument Delphi.

VII. CONCLUSION

This study is first time a computerized version of argument Delphi—a variant of policy Delphi—has been proposed. The technique is developed for several alternative cases and the problems faced during the application of the method to real life problems and solutions are proposed and applied. This novel version of the Delphi technique can be applied to many environments, including Web 3.0, like social networks, web platforms, blogs, forums and wikis [40].

Besides the proposed processing techniques on the argument graph, introduced for the first time during the process of the computerized argument Delphi method, it is believed that the graph model of argument Delphi will be an acceleration of current studies and it is open for further studies of graph theory, statistics and social sciences.

One natural outcome of the real-life problems yielded 40 alternative implementations of the computerized argument Delphi method as a variation of authorized/unauthorized versions or argument aggregation by "list of arguments", "drop similar", "tournament selection" or "use qualitative marginal" or the evaluation of arguments by "sort and rate", "propagated path sort", "average propagation sort", "leave lowest out" or "two camps approach". All these methods of computerized argument Delphi are here introduced into the literature for the first time. In addition, the number of arguments in each step or the final step is left parametric for the facilitator.

REFERENCES

[1] N. Dalkey and O. Helmer, "An experimental application of the DELPHI method to the use of experts," *Manage. Sci.*, vol. 9, no. 3, pp. 458–467, Apr. 1963.

[2] H. A. Linstone and M. Turoff, *The Delphi Method: Techniques and Applications*. Reading, MA, USA: Addison-Wesley, 1975.

[3] R. L. Custer, J. A. Scarcella, and B. R. Stewart, "The modified Delphi technique—A rotational modification," *J. Vocational Tech. Edu.*, vol. 15, no. 2, pp. 50–58, 1999.
[4] G. D. Garson, *The Delphi Method in Quantitative Research* (Statistical Associates 'Blue Book' Series Book), vol. 44. Asheboro, NC, USA: Statistical Associates Publishers, 2013.
[5] S. Keeney, H. McKenna, and F. Hasson, *The Delphi Technique in Nursing and Health Research*. New York, NY, USA: Wiley, 2011.
[6] R. J. Heuer, Jr., and R. H. Pherson, *Structured Analytic Techniques for Intelligence Analysis*. Washington, DC, USA: CQ Press, 2014.
[7] B. K. Eakman, *How to Counter Group Manipulation Tactics: The Techniques of Unethical Consensus-Building Unmasked*. Raleigh, NC, USA: CreateSpace Independent Publishing Platform, 2003.
[8] M. I. Yousuf, "The Delphi technique," *Essays Edu.*, vol. 20, pp. 80–89, 2007.
[9] W. G. Rieger, "Directions in Delphi developments: Dissertations and their quality," *Technol. Forecasting Social Change*, vol. 29, no. 2, pp. 195–204, Apr. 1986.
[10] D. Gallego and S. Bueno, "Exploring the application of the Delphi method as a forecasting tool in information systems and technologies research," *Technol. Anal. Strategic Manage.*, vol. 26, no. 9, pp. 987–999, Jul. 2014.
[11] K. Kauko and P. Palmroos, "The Delphi method in forecasting financial markets—An experimental study," *Int. J. Forecasting*, vol. 30, no. 2, pp. 313–327, Jun. 2014.
[12] M. Adler and E. Ziglio, Eds., *Gazing Into the Oracle: The Delphi Method and Its Application to Social Policy and Public Health*. London, U.K.: Jessica Kingsley Publishers, 1996.
[13] R. D. Needham and R. C. de Loë, "The policy Delphi: Purpose, structure, and application," *Can. Geographer*, vol. 34, no. 2, pp. 133–142, Jun. 1990.
[14] C. L. Paul, "A modified Delphi approach to a new card sorting methodology," *J. Usability Stud.*, vol. 4, no. 1, pp. 7–30, Nov. 2008.
[15] M. Turoff and S. R. Hiltz, "Computer based Delphi processes," in *Gazing Into the Oracle: The Delphi Method and Its Application to Social Policy and Public Health*, M. Adler and E. Ziglio, Eds. London, U.K.: Jessica Kingsley Publishers, 1996, pp. 56–88.
[16] O. Kuusi, *Expertise in the Future Use of Generic Technologies: Epistemic and Methodological Considerations Concerning Delphi Studies*. Helsinki, Finland: Unigrafia Bookstore, 1999.
[17] K. Korhonen-Yrjänheikki, "Using argument Delphi to anticipate the challenges of engineering education in a systematic way: Case Finland 2015," in *Proc. SEFI Annu. Conf.* , Valencia, Spain, 2004, pp. 221–227.
[18] S. E. Seker, C. Mert, K. Al-Naami, N. Ozalp, and U. Ayan, "Time series analysis on stock market for text mining correlation of economy news," *Int. J. Soc. Sci. Humanity Stud.*, vol. 6, no. 1, pp. 66–91, 2013.
[19] S. E. Seker, K. Al-Naami, and L. Khan, "Author attribution on streaming data," in *Proc. IEEE 14th Int. Conf. Inf. Reuse Integr. (IRI)*, Aug. 2013, pp. 497–503.
[20] R. Loo, "The Delphi method: A powerful tool for strategic management," *Policing, Int. J. Police Strategies Manage.*, vol. 25, no. 4, pp. 762–769, 2002.
[21] A. R. McMurray, "Three decision-making aids: Brainstorming, nominal group, and Delphi technique," *J. Nursing Staff Develop.*, vol. 10, no. 2, pp. 62–65, Mar./Apr. 1994.
[22] A. M. Kaplan and M. Haenlein, "Users of the world, unite! The challenges and opportunities of social media," *Bus. Horizons*, vol. 53, no. 1, pp. 59–68, Jan./Feb. 2010.
[23] S. Yardi and D. Boyd, "Dynamic debates: An analysis of group polarization over time on Twitter," *Bull. Sci. Technol. Soc.*, vol. 30, no. 5, pp. 316–327, Oct. 2010.
[24] D. A. Shamma, L. Kennedy, and E. F. Churchill, "Tweet the debates: Understanding community annotation of uncollected sources," in *Proc. 1st SIGMM Workshop Social Media (WSM)*, Beijing, China, 2009, pp. 3–10.
[25] S. E. Seker and K. Al-Naami, "Sentimental analysis on Turkish blogs via ensemble classifier," in *Proc. Int. Conf. Data Mining*, 2013, pp. 10–16.
[26] S. E. Seker, "Sentimental versus impact of blogs," in *Proc. Int. Conf. Internet Comput. Big Data (ICOMP)*, 2013, pp. 127–133.
[27] G. W. F. Hegel, F. H. Simon, and T. Rockmore, *Hegel's Lectures on the History of Philosophy*. Amherst, NY, USA: Prometheus Books, 1996.
[28] S. E. Seker, O. Altun, U. Ayan, and C. Mert, "A novel string distance function based on most frequent K characters," *Int. J. Mach. Learn. Comput.*, vol. 4, no. 2, pp. 177–183, 2014.
[29] S. E. Seker and C. Mert, "A novel feature hashing for text mining," *J. Tech. Sci. Technol.*, vol. 2, no. 1, pp. 37–40, Jun. 2013.
[30] B. L. Miller and D. E. Goldberg, "Genetic algorithms, tournament selection, and the effects of noise," *Complex Syst.*, vol. 9, no. 3, pp. 193–212, 1995.
[31] B. Satyanarayana and K. S. Prasad, *Discrete Mathematics and Graph Theory*. Cambridge, MA, USA: PHI Learning Pvt. Ltd., 2006.
[32] W. McCuaig, "Intercyclic digraphs," in *Proc. AMS-IMS-SIAM Joint Summer Res. Conf. Graph Minors Graph Struct. Theory*, 1992, p. 205. [Online]. Available: http://books.google.com/books?id=idigH5CTGWAC&pg=PA205
[33] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," in *Proc. 7th Int. Conf. World Wide Web*, 1998, pp. 107–117.
[34] J. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, no. 5, pp. 604–632, Sep. 1999.
[35] K. Bryan and T. Leise, "The $25,000,000,000 eigenvector: The linear algebra behind Google," *SIAM Rev.*, vol. 48, no. 3, pp. 569–581, 2006.
[36] G. Ertek and A. Demiriz, "A framework for visualizing association mining results," in *Computer and Information Sciences—ISCIS* (Lecture Notes in Computer Science), vol. 4263. Berlin, Germany: Springer-Verlag, 2006, pp. 593–602.
[37] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," in *Proc. 3rd Int. AAAI Conf. Weblogs Social Media AAAI Pub.*, 2009, pp. 361–362.
[38] R. Diestel, *Graph Theory*, 3rd ed. New York, NY, USA: Springer-Verlag, 2005.
[39] F. W. Takes and W. A. Kosters, "Computing the eccentricity distribution of large graphs," *Algorithms*, vol. 6, no. 1, pp. 100–118, 2013.
[40] M. L. Arslan, S. E. Seker, and C. Kızıl, "Innovation driven emerging technology from two contrary perspectives: A case study of Internet," *Emerg. Markets J.*, vol. 3, no. 3, pp. 87–97, Mar. 2014.

SADI EVREN SEKER (M'01) received the B.Sc., M.Sc., and Ph.D. degrees from the Computer Science and Engineering Department. He was a Post-Doctoral Researcher in streaming data mining and social networks with the Computer Science Department, University of Texas at Dallas. He has studied in natural language processing and artificial intelligence subjects during his education. He is currently an Academic Staff with the Management Information System Major Program, Business Department, Istanbul Medeniyet University. He has authored numerous peer-reviewed papers, books, and patents. His active research topics are business intelligence, data mining, and social networks.