Opinion Discovery Framework: Toward a Quality Opinion-Centric Platform

Najla Althuniya, Xiaoqing Liu, Joseph W. Sirrianni, and Douglas Adams
University of Arkansas, Fayetteville, AR, USA
Email: {nalthuni, frankliu, jwsirria, djadams}@uark.edu

***Abstract*—People use argumentation and deliberation platforms to express and share opinions. As a result, these platforms contain massive opinions where users cannot keep track of or identify where constructive opinions are. In our study, we found that users only view, on average, 3% of the available content. Usually, opinions are discovered by engagement information, impact score or reverse-chronological order but opinions contain substantial information beyond its text. Features for opinions discovery improves the discourse quality and provides an attractive online discussion platform. However, there are no clear feature sets for the opinions that have been identified for searching and discovering opinions in academia or public-debate platforms. This paper proposes a novel innovative framework for opinion selection and discovery. It discovers constructive opinions based on four unique features: engagement, recentness, controversy, and author influence. Therefore, it provides a dynamic discourse incorporating opinion's features based on users' preferences. We first defined those features in the cyber-argumentation space. Then, we discuss our new framework that combines those features for opinion search and discovery. An application example on a deliberation dataset has shown that our framework works effectively on discovering and searching constructive opinions.**

***Index Terms*—argumentation, engagement, recentness, search algorithm, search, opinion, controversy, author influence, discovery**

I. INTRODUCTION

User-generated content (UGC) platforms, such as social media, blogs, photo sharing, and websites, allow extensive discussion and participation from users. Generally, UGC content is created by regular people and mostly unsanitized as traditional media outlets. Therefore, UGC content suppliers are rewarded by receiving recognition from content consumers, who use UGC platforms for information or entertainment. As a result, UGC applications create an attractive user environment and adapted AI models to help users be more active, creative, and develop new personal and business opportunities. Cyber-argumentation systems, an AI sub-field, are an example of UGC platforms. Online-Argumentation platforms allow vast discourse between participants as well as understanding the discussion. It addresses issues by creating well-defined structures for deliberation. It has exhibited the ability to evaluate the discussion on large-scale platforms and in different contexts [1]-[4].

While large-scale deliberation systems host massive opinions, not all opinions are worth further discussion. Opinions in cyber-argumentation systems hold rich information beyond its text. Commonly, in public deliberation systems [5], [6], opinions can be discovered by engagement information, impact score, or reverse-chronological order. These discovering methods omit important features an opinion may exhibit. They degrade some constructive opinions using only one feature for opinions search and discovery. Constructive opinions, or critiques, usually hold good intentions, motives for improvements and positive feedback that makes a particular situation better. Therefore, identifying features of constructive opinions for searching and discovering opinions is needed. However, it is a challenging task because there are no clear definitions and measurements for opinion features. Some opinions are informative, related, controversial, or none of the previous. For instance, constructive opinions receive more attention over time than unconstructive opinions. It also creates a controversial state over the written argument, which attracts the audience to react and engage with them. Thus, constructive opinions retain some unique characteristics. Recognizing the different attributes of worthwhile opinions provide signals to what degree each opinion should be promoted and presented to users. Discovering and promoting opinions using different opinion's features will lead to higher quality discussions because it will take into account various dimensions of searching opinions instead of using only one dimension for searching opinions. Constructive discourse results in strengthening online communities and uniting users [7]. Therefore, users reach their best-reasoned judgment to solve a problem, increase users' participation in the discussion and achieve the argumentation system goals.

In this paper, we present a novel selection framework for searching and discovering constructive opinions in cyber-argumentation platforms. This framework searches and discovers opinions based on four non-textual opinion features to create a dynamic opinion-centric platform. This framework is a user-customized search and uses four opinion-distinguishing features: engagements, recentness, author influence, and controversy. We first define these features in the cyber-argumentation field, analyze the relationship between them, and then we introduce our

Manuscript received February 12, 2020; revised April 13, 2020.

doi: 10.12720/jait.11.2.48-57

new framework. We used our dataset collected from our argumentation tool and our framework to create a dynamic opinion discovery system.

This paper is organized as follows. Section II introduces argumentation systems and our argumentation platform. Section III refers to some of the work done in opinions' discovery and selection in research and public debate platforms. Section IV describes the opinions' discovery and selection problem in our argumentation system. Section V introduces opinions features, the proposed framework. Then, Section VI demonstrates our dataset used in this research. Finally, Section VII illustrates an example of different argument discovery methods.

II. ARGUMENTATION SYSTEMS

The cyber-argumentation systems, a sub-field of AI, refer to online cyber-tools that allow participants to share and discuss ideas with each other. These platforms are developed to facilitate online discussion effectively and provide structured frameworks for argumentation and deliberation. Besides, these argumentation tools are capable of identifying groupthink [2], analyzing argument credibility [1], measuring polarization in opinions [3], recommending friendship connections using opinion diversity [8] and predicting the missing collective opinion [4]. Usually, argumentation systems are built on formal models such as Dung abstract frameworks [9], informal models such as IBIS [10] or some combination of models.

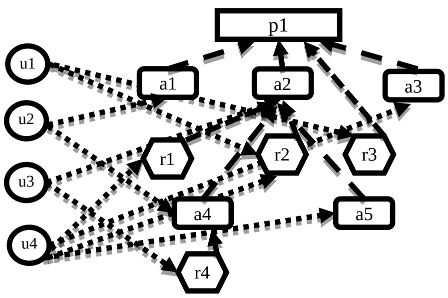

Figure 1. User / Position / Argument tree

In this work, we use our argumentation platform, the Intelligent Cyber Argumentation System (ICAS). ICAS is a web-based argumentation platform based on the informal Issue-Based Information System model [10]. The structure of ICAS can be represented as multiple weighted directed trees. Each issue can be considered as a tree root. Issues are the unsolved problem or open questions for discourse. To solve the problem represented by an issue, positions are suggested. Positions serve as the stances, viewpoints or solutions for an issue. Each position addressing that issue is a child to the root (issue). Users can support or attack each position with opinions (hence, we refer to opinions here as arguments), as in Fig. 1. Additionally, users can support and attack each other's arguments. Arguments are users' opinions for or against positions or other arguments. These arguments become children to a position (in this case, we call them first-level arguments) or other arguments (in this case, they are called counter-arguments). Each argument has a text and an agreement value. If users want to support or attack positions or arguments without text, they can react to arguments or positions. Reactions are a user-defined value that indicates the user's agreement or disagreement on the parent node. The agreement value is a double number ranging from -1 to 1, where the sign indicates the user's agreement or disagreement with the parent node and the value represents the intensity of agreement/disagreement. For example, a user may post (-0.6) as an agreement value. Because the sign is negative, it is a disagreement. The 0.6 indicates that the user is strongly disagreeing. For more information about the ICAS structure, please refer to [11].

III. RELATED WORK

Limited research has been done on argument search and discovery. Nguyen *et al.* 2017 has developed an end-end process in argument discovery to minimize the crowd cost and maximize the quality of crowd answers using argument text [12]. Mishne, 2006 identified opinions in blogs by scoring posts based on various aspects associated with an expression of opinion about a topic. It included shallow sentiment analysis, spam detection, and link-based authority estimation [13]. Amgoud & Ben-Naim, 2013 proposed a new family of semantics, which searches and rank-orders arguments from the most acceptable to the weakest one. Thus, their approach offers a theoretical framework for comparing semantics [14]. While Bonzon *et al.* 2016 have proposed six new ranking-based semantics for search results based on the propagation of the weights of arguments, their approach gave a higher weight to non-attacked arguments [15]. Moreover, Eirinaki *et al.* 2012 presented an algorithm that analyzes the overall sentiment of a document/review, and identifies the semantic orientation of specific components of the review that lead to a particular sentiment [16]. Finally, Pu *et al.* 2014 defined new ranking-based semantics, called categoriser-based ranking semantics, for abstract argumentation framework. All the above-mentioned research-work focuses on discovering arguments using textual information as the foundation for argument discovery methods [17].

Moreover, there are some public debate platforms such as Kialo [5] and Debate.com [6] where opinions are discovered differently. For example, in Kialo, opinions under the thesis statement are searched and listed using the impact score. The opinions are listed in decreasing order based on their impact score. The impact scores are calculated using the users' ratings; a user can rate an opinion impact on a scale [1-5], and other information. Likewise, in Debate.com, users can choose from different discovery methods to sort out the opinion theses such as by date, most agreed, most disagreed and unanswered. Then, under each opinion, the counter-opinions are listed based on number of users' interactions with those opinions such as replies and likes. In all the above-mentioned platforms, the temporal information has been omitted. In addition to that, the opinion author's status has not been considered for scoring or discovering those opinions. Furthermore, there was no consideration for controversy in their discovery methods.

On other UGC platforms, there are many aspects to consider for scoring or searching posts. In particular, social media applications have rich content from users. This content is processed to bring the most attractive and related content to users. For instance, Facebook [18], the most popular platform for social connections, has developed a personalized news feed algorithm to rank user's stories that matter most to a user every time he or she visits Facebook. It searches and evaluates each story based on who posted it, what kind of media it contains, and interactions made so far to each story. It scores each story based on that information and places the ones that matter the most to users on the top of the feed. Another UGC example is Twitter [19]. Twitter is a popular microblogging platform. It categorizes each user's timeline into three sections. The first section displays the top tweets using tweets engagement, user's connection, and user preferences information while the final section lists the remaining tweets and events in reverse-chronological order. Each UGC platform has its design and attributes. Considering those attributes to evaluate content is a significant reason for platform success and attracting more users.

With the expansion of UGC platforms and the advance in AI techniques, cyber-argumentation platforms are becoming widely available and supported. Cyber-argumentation platforms differ from other UGC platforms by capturing the crowd wisdom and collective opinions dynamically. To help to achieve this goal, there is a need to develop an opinion discovery framework toward a quality opinion-centric platform. This paper proposes a new opinion discovery framework that uses a different non-textual opinion's features to create a quality opinion centric platform.

IV. Problem Description

ICAS has more than 10600 arguments spread among 16 positions. On average, there are 408 first-level arguments for each position. From our study, on average, users view only 3% of the content. Users do not explore all arguments due to the massive content and time limitation. Therefore, we need to select and present the most constructive arguments for users. However, this is a challenging task for many reasons. First, this dataset has large-scale discussions, which stresses the need for argument discovery framework. Therefore, users have no time to go through all posted arguments to find the constructive arguments. Secondly, users have no choice to select or express their preferences. Finally, there are no clear argument features set that we can use to choose from to build the arguments list. Arguments can be old, new, engaging, not engaging, short, long, reasonable, sound, valid, controversial, docile, etc. Arguments' features are either explicit, such as created date, number of reactions, agreement value, etc., or implicit, such as the degree of engagement, controversy, etc. ICAS searches and sorts arguments by reverse-chronological order, which is the common argument discovery method for deliberation platforms. Therefore, all new arguments will always be at the top of the list, pushing old arguments down the list. This searching method is reasonable and commonly used but it has limitations. First, it is not capable of recognizing the old arguments with recent user engagement. Therefore, these arguments receive less or no user engagement over time. Secondly, ordering the arguments list by date cannot capture the other significant arguments' features. However, not all arguments are engaging. In our dataset, more than half of the arguments received no engagement from users. Discovering arguments by engagement information gives more exposure to arguments with higher user engagement regardless of the other features. Therefore, the new arguments are penalized for appearing further down on the search list. Failing to recognize the temporal aspect of an argument can negatively affect the user experience, and make the arguments list to appear unchanged. Another vital feature for arguments is the degree of controversy. Some arguments are controversial, making users reacting with different levels of agreement or disagreement. Identifying the controversy degree for arguments is the main core of the argumentation systems. Finally, the last feature that plays an important role in argument discovery is the argument's author's influence. People come with different knowledge and skills. Arguments' authors have a direct or indirect impact on the readers' thoughts, feelings, and actions. Failing to recognize the authors' influences aspect of an argument can undermine its significance. Up to our knowledge, there is no research that has been done so far that considers the above-mentioned arguments' features for argument discovery. Hence, we are referring to these features as indicators for the rest of this research.

V. Proposed Framework

According to the multi-faceted concept of argument discovery, we propose a new framework to search and discover the most constructive arguments in a position tree based on four non-textual indicators: Argument's Engagement, Argument's Recentness, Argument's Degree of Controversy and Arguments Author's Influence Degree (hence, we refer to the proposed framework as the Argument Discovery Framework). We define constructive arguments as new engaging arguments that create a controversial state written by an influential. Therefore, we are focusing on the above-mentioned indicators to discover the most constructive opinions in cyber-argumentation platforms.

In this section, we first define the argument indicators: engagement, recentness, controversy, and author influence and methods to quantify them in the cyber-argumentation space. Then, we analyze the correlations between the indicators to apply the recommended scoring method. Finally, we introduce the aggregated argument discovery framework.

A. Preliminaries

An argument is made up of related arguments and reactions. An argument is defined in ICAS as a tuple:

<*a*, *p*, *u*>

To create an argument, there are three entities involved while submitting a new argument process:

- *a* ∈ A: is the argument to be made. It has some features such as ID, text, agreement value, created date, last activity time.
- *p* ∈ {A, P}: is the parent node for the made argument. *p* could be an argument ∈ A, or a position ∈ P. It has the same features as the argument.
- *u* ∈ U: is the user who authored the argument.

Similarly, the reaction is defined in ICAS as a tuple:

<*r*, *p*, *u*>

To create a reaction, there are three entities involved while submitting a new argument process:

- *r* ∈ R: is the reaction to be made. It has some features such as ID, agreement value, created date.
- *p* ∈ {A, P}: is the parent node for the made argument. *p* could be an argument ∈ A or a position ∈ P. It has the same features as the argument.
- *u* ∈ U: is the user who posted the reaction.

It is intuitive to construct a graph to illustrate the involved factors in argumentation behavior. The graph is the argumentation graph as shown in Fig. 1. Now we give a formal definition of the argumentation graph.

Definition 1. (Argumentation Graph) argumentation behavior can be represented as a graph AG = (V, E) where:

- V = A ∪ P ∪ R ∪ U. Four types of entities are involved: A arguments, P positions, R reactions and U users.
- E = <*u*, *p*> ∪ <*u*, *a*> ∪ <*u*, *r*> ∪ <*a*, *p*> ∪ <*r*, *p*> ∪ <*a*, *a*> ∪ <*a*, *r*>. <*u*, *p*> represents *u* posted a position *p*. <*u*, *a*> represents *u* posted an argument *a*. <*u*, *r*> represents *u* posted a reaction *r* to an argument or position. <*a*, *p*> represents an argument *a* posted to a position *p*. <*r*, *p*> represents *r* a reaction made to a position *p*. <*a*, *a*> represents an argument *a* posted to an argument *a*. <*r*, *a*> represents *r* a reaction made to an argument *a*.

The argumentation graph serves as the baseline graph for this research work. This graph is used and modified in many ways to extract extra information used in this research.

B. *Engagement*

Usually, user engagement is defined as a quality of user experience with technology [20]. In ICAS, users can be engaged with the system by viewing or adding issues, positions or arguments, update an argument, and react to an argument or position and other functions. All user's interactions made to issues, positions or arguments are considered as an engagement. Therefore, we can measure the engagement for an entity by the amount and the kind of interactions made by users to that entity. Users can react or reply to an argument or position. A user's reaction is determining the level of agreement/ disagreement with the parent entity without text. A user's reply is determining the level of agreement/ disagreement with the parent entity with text. Therefore, replies have more weight than reactions in an entity engagement measurement because it contains additional information than reactions. To measure the degree of engagement for each argument, we use some data sources from the AG:

- V = A ∪ P ∪ R. Three types of entities are involved: A arguments, P positions, and R reactions.
- E = <a, p> ∪ <r, p> ∪ <a, a> ∪ <r, a>. <a, p> represents an argument *a* posted to a position *p*. <r, p> represents *r* a reaction made to a position *p*. <a, a> represents an argument *a* posted to an argument *a*. <r, a> represents *r* a reaction made to an argument *a*.

We assume that engagement information contributes to:

- Node Weights: Node weights can be interpreted as the sum of all edges' weights connected to this node.
- Edge Weight: Edge Weight is obtained from the relation between entities. If the relation is created between two arguments, that edge weighs two. If the relation is created between a reaction and an argument, that edge weighs one.

According to the above assumptions, the argument is defined as a tuple:

<*a*, *p*, e_{ap}>

Similarly, the reaction is defined as a tuple:

<*r*, *a*, *p*, e_{rp}>

To calculate an argument engagement in AG, there are two entities involved during submitting a new argument process:

- *a*, *r*, *a*, *p*: the same as in the AG.
- e_{ap}: is the engagement weight assigned to the created relation between the argument *a* and every ancestor in the path of argument *a*. The weight is equal to two because each argument is made up of an agreement value and a text.
- e_{rp}: is the engagement weight assigned to the created relation between the reaction *r* and every ancestor in the path of reaction *r*. The weight is equal to one because each reaction is made up of an agreement value only.

The total engagement score $TE(a_i)$ can be calculated for each argument as:

$$\boldsymbol{TE(a_i)} = \sum_{r_j, a_j \to a_i} \boldsymbol{w_{ij}} \tag{1}$$

where

$$w_{ij} = \begin{cases} 2, & \text{if } a_j \text{ has a relation with } a_i \text{ in } AG \\ 1, & \text{if } r_j \text{ has a relation with } a_i \text{ in } AG \\ 0, & \text{else} \end{cases} \tag{2}$$

To measure the degree of engagement for argument *a*, we divide the total engagement of argument *a* by the sum of the total engagement of all *a*'s siblings at the same level. The engagement score for argument *a* is calculated as:

$$\mathbf{E\_score}(a_i) = \frac{\boldsymbol{TE(a_i)}}{\boldsymbol{Max}\left(\sum_{a_j \in A} \boldsymbol{TE(a_j)}, \mathbf{1}\right)} \tag{3}$$

To avoid dividing by zero, we added the maximum function to choose between the maximum total engagement information and one. This smoothing to take into account arguments or positions without children. The formula (3) returns a double value [0, 1], where 0 means the argument has no engagement at all and 1 means the argument is the most engaging argument.

C. Recentness

Recentness can be interpreted in different ways in ICAS. For specific temporal actions, such as posting new arguments, this is considered as recent activity. On the other hand, partial update to arguments such as updating content, receiving new counter-arguments or reactions is considered as a recent update. For both situations, the argument time information has changed. Specifically, in the second situation, the argument tree has expanded and received either more information, more support, or attack. Therefore, we can measure the recentness for each entity by passing on the entity's last activity time to all the entity's ancestors. To measure the degree of recentness for each entity, an argument or position, we use the same data sources used to measure the engagement information. To calculate an argument recentness in AG, there are two entities involved during submitting a new argument process:

- a, r, p: the same as in the AG.
- r_{ap}: is the recentness weight assigned to the created relation between argument *a* and every ancestor in the path of argument *a*.
- r_{rp}: is the recentness weight assigned to the created relation between reaction *r* and every ancestor in the path of reaction *r*.

To calculate the recentness score for argument *a*, we need to find the recentness information for each argument and the maximum recentness information among its siblings. However, the recentness information for arguments is composed of date and time information. Therefore, we need to normalize the last activity time for all arguments into an interval of time units, such as minutes, hours or days, to find the maximum recentness. This normalization will provide us with an interval of time units. The interval's endpoints can be set to thresholds. For example, the upper endpoint can be set to the current time, and the lower endpoint can be set to a few hours or days early. In this study, we use days as interval units. We normalized the argument's last activity time (date only) to an interval of integers. The normalized start date of the study is used as the lower endpoint for the interval. The normalized last date of the study is used as the upper endpoint for the interval. The recentness information for each argument is calculated as follows:

$$\boldsymbol{R}(a_i) = \boldsymbol{normalized}\ (\boldsymbol{last\ activity\ time}(a_i)) \tag{4}$$

The maximum recentness information is the maximum value of recentness information from all arguments with the same level as a_i. After that, calculate the recentness score for each argument as:

$$\boldsymbol{R\_score}(a_i) = \frac{\boldsymbol{R}(a_i)}{\boldsymbol{Max}\ (\boldsymbol{R}_{a_j \in A}(\boldsymbol{a_j}), \boldsymbol{1})} \tag{5}$$

Similar to the engagement score, smoothing is applied for the recentness score. The formula (5) returns a double value [0, 1], where 0 means the argument is an old argument and 1 means the argument is new.

D. Controversy

The controversy is a major phenomenon in argumentation systems. However, there is no unified definition for controversy in UGC platforms. Dori-Hacohen (2017) quantifies controversy as the degree of disagreement among large groups of people in discussion or issues [21]. Garimella *et al.*, (2018) quantifies controversy on the topic's level and the user's level using different methods [22]. In ICAS, the argument's controversy measures the degree of users' agreement or disagreement for a position or an argument. Argument *a* with constant supports means that users agreed on supporting and accepting this argument. Argument *a* with constant attacks means that users agreed on disagreement or rejection of this argument. However, argument *a* with different levels of supports and attacks is more controversial. Therefore, to measure the degree of controversy for argument *a*, we use some data sources from AG, the same sources used to define arguments' engagement and recentness. We use the agreement value of an argument or reaction as the base information. Agreement values range [-1, +1]. (-1) means a strong disagreement or attack and (+1) means strong agreement or support. We used the standard deviation to measure the degree of controversy for an argument or a position. Usually, the standard deviation of data points is frequently used as a measure of the volatility of those points. If all data points are closer to the mean, the standard deviation is low. If the data points are spread out over a wider range of values, then the standard deviation is high. Therefore, an argument with constant attacks or supports will have a low standard deviation. It means that the supporters or attackers are in agreement. However, arguments with mixed attacks or supports will have a higher standard deviation. We can capture this disagreement between people by calculating the standard deviation on the agreement values made to an argument. The controversy degree for argument *a* is calculated for all direct arguments' and reactions' agreement values made to argument *a* as:

$$\boldsymbol{C\_score}(\boldsymbol{a}) = \sqrt{\frac{\boldsymbol{1}}{\boldsymbol{N-1}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} (\boldsymbol{x_i} - \overline{\boldsymbol{x}})^2} \tag{6}$$

x_i is the agreement value of a direct child for argument *a*, $\overline{x}$ is the mean value of argument *a* direct children agreement values, *N* is the number of direct children to argument *a*. For siblings arguments, the C_score(a) gets normalized to [0, 1] where 0 means the argument has a low controversy degree and 1 means the argument has a high controversy.

E. Author's Influence

This indicator determines a user's influence on the discussion. Users in the discussion are authors, readers or both. We are interested in the author's influence. When a user posts an argument and he gets many replies and reactions, he influences the responders. This could happen in a different context. A user may have a strong influence in one or more positions and no influence on the other positions. Therefore, we are calculating the degree of author influence on the position level. We used the AG to create the User Interaction Graph (UIG) in Fig. 2. Now we give a formal definition of the User Interaction Graph.

Definition 2. (User Interaction Graph) users' interaction behavior can be represented as a graph UIG = (V, E) where:

- V = U, only one type of entity, which is users U.
- E = <u_i, u_j> represents that u_i has replied or reacted to an augment written by u_j.

We assume that author influence information contributes to:

- Node Weights: Node weights can be interpreted as the degree of the user's influence on the discussion.
- Edge Weight: Edge Weight is obtained from the relation between users. If u_i has replied to an augment written by u_j, that edge weighs two. If u_i has reacted to an augment written by u_j, that edge weighs one.

We used the UIG information as the data sources to calculate the degree of author influence in the discussion. PageRank Algorithm [23] has been widely used to identify influentials in different scenarios [24], [25]. To measure the author's influence for an argument *a*, we used the weighted PageRank Algorithm [23].

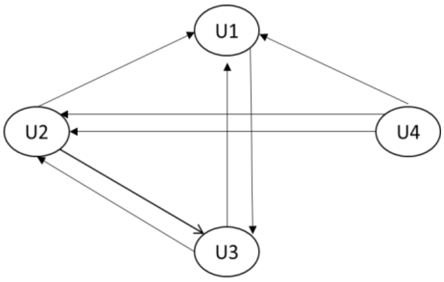

Figure 2. User Interaction Graph (UIG)

$$\boldsymbol{AI\_score}(\boldsymbol{a}_{\boldsymbol{u}_i}) = (\mathbf{1}-\boldsymbol{d}) + \boldsymbol{d} \sum_{u_i \to u_i} \boldsymbol{PR}(\boldsymbol{u}_j)\, \boldsymbol{W}^{\boldsymbol{in}}_{(\boldsymbol{u}_j,\boldsymbol{u}_i)}\, \boldsymbol{W}^{\boldsymbol{out}}_{(\boldsymbol{u}_j,\boldsymbol{u}_i)} \quad (7)$$

$W^{in}_{(u_j,u_i)}$ is the weight of link (u_j, u_i) calculated based on the number of in links of user u_i and the number of all references users of user u_j. $W^{out}_{(u_j,u_i)}$ is the weight of link (u_j, u_i) calculated based on the number of out links of user u_i and the number of all references users of user u_j. The $PR(u_i)$ score range is [0, 1], where 0 mean user u_i has a low influence degree and 1 means user u_i has a high influence degree as an author.

F. *Correlations between Indicators*

We have performed the PEARSON correlation [26] between the above-mentioned indicators for all positions' discussions in ICAS. Fig. 3 shows us the correlations between the four indicators.

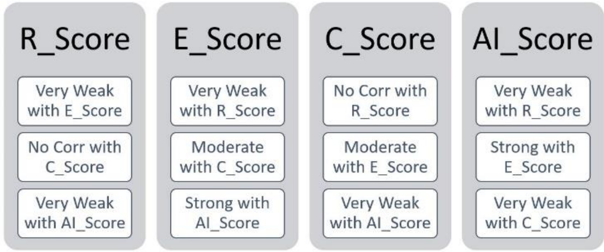

Figure 3. Correlation between arguments indicators

According to [26], the recentness indicator (R_Score) has no correlation with the controversy indicator (C_Score) and a very weak correlation with the other indicators. The reason behind this is that, over time, users lose interest in viewing and interacting with old postings. However, the engagement indicator (E_Score) has a moderate correlation with the controversy indicator (C_Score) and a strong correlation with the author's influence indicator (AI_Score). The reason behind those correlations is that these scoring methods use the same sources of data to score the arguments for those indicators. Although the controversy indicator (C_Score) and the author influence indicator (AI_Score) use the same source of data for scoring arguments, they have a very weak correlation with each other. In other words, a controversial argument does not have to be written by an influential author and vice versa. In order to find the relation between any indicators, we need to perform the PEARSON correlation [26] between them. Then, based on the *r*-value, we can use the appropriate operator to combine those indicators. We analyzed the relationships between the indicators to help us build the argument discovery framework.

a) Relationships among indicators

We found that some of the indicators have a strong correlation with each other such as the engagement indicator (E_Score) and the controversy indicator (C_Score). However, there are moderate, weak or no correlations that exist between the indicators as shown in Fig. 3. Thus, we are categorizing the relationships between the indicators as in [27] into conflicting, cooperative, and mutually exclusive.

i) Conflicting indicators ($\otimes$)

According to Liu *et al.*, (2012), two indicators are said to be conflicting if the correlation value $r < 0$ [27]. If there is an increase in one indicator, it always leads to a decrease in the other indicator. Therefore, they are completely conflicting. In our setting, there are no conflicting indicators. However, if this framework gets expanded or modified in the future, it might have conflicting indicators. In this situation, for each argument *a*, we apply the fuzzy compromise operator $\otimes$ on the argument indicators to score argument *a*. Consider the following set of indicators scores for an argument *a*: $\{I_1(a),\ I_2(a),\ \dots\ I_n(a)\}$. This operator combines all included indicators and uses the average function to trade off between those conflicting indicators as follows:

$$\boldsymbol{I_1(a)} \otimes \boldsymbol{I_2(a)} = \frac{\boldsymbol{I_1(a)} + \boldsymbol{I_2(a)}}{\mathbf{2}} \quad (8)$$

ii) Cooperative indicators (⊕)

According to Liu *et al*., (2012), two indicators are said to be cooperative if the correlation value $r > 0$ [27]. If there is an increase in one indicator, it always leads to an increase in the other indicator. Therefore, they are satisfied at the same time. An example is a correlation between the engagement (E_Score) and the controversy (C_Score) indicators. In this situation, for each argument *a*, we apply the fuzzy conjunction operator ⊕ to score arguments. Consider the following set of indicators scores for an argument *a*: $\{I_1(a), I_2(a), \ldots I_n(a)\}$. This operator combines all included indicators and uses the MIN function [27] for combining those cooperative indicators to get services with the cheapest price. However, in our context, the MIN function is not suitable because it would degrade the argument score. Because we are trying to find the arguments with the highest score and the indicators are cooperative, we are multiplying the indicators scores to score argument *a* as follows:

$$\boldsymbol{I_1(a) \oplus I_2(a) = I_1(a) * I_2(a)} \tag{9}$$

iii) Mutually exclusive indicators (⊙)

According to Liu *et al*., (2012), two indicators are said to be mutually exclusive if there are no correlations between the indicators [27]. Therefore, they cannot be satisfied at the same time to the highest degree. An example is a correlation between the recentness (R_Score) and the controversy (C_Score) indicators. In this situation, for each argument *a*, we apply the fuzzy disjunction operator ⊙ to score arguments. Consider the following set of indicators scores for an argument *a*: $\{I_1(a), I_2(a), \ldots I_n(a)\}$. This operator combines all included indicators and uses the MAX function to combine those mutually exclusive indicators as follows:

$$\boldsymbol{I_1(a) \odot I_2(a) = MAX(I_1(a), I_2(a))} \tag{10}$$

G. The Argument Discovery Scoring Model

Not all users are willing to choose the argument discovery method. Moreover, we do not expect users to understand the arguments indicators and their correlations, especially when the recentness indicator has different correlations values with the other indicator. That might cause the users not to be able to build the recommended argument discovery method. Yet, we want to discover the most constructive arguments for users. Therefore, we build the argument discovery model. This model aggregates the argument indicators while paying equal attention to each of the four indicators and uses a Structured Equation Modeling (SEM) technique [28] to build a linear model to synthesize the argument's indicators. SEM is frequently used to evaluate and assess unobservable 'latent' constructs using one or more observed variables. The argument discovery score method is formulated using SEM as follows (Fig. 4):

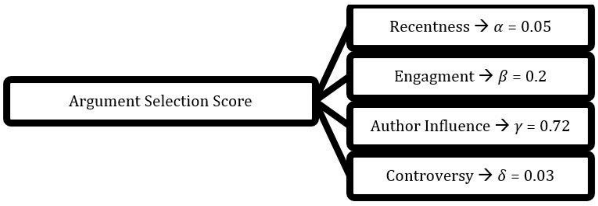

Figure 4. Argument discovery scoring model

$$\mathbf{S(a)} = (\boldsymbol{\alpha} * \mathbf{R\_score(a)}) + (\boldsymbol{\beta} * \mathbf{E\_score(a)}) + (\boldsymbol{\gamma} * \mathbf{PR(u_a)}) + (\boldsymbol{\delta} * \mathbf{C\_score(a)}) \tag{11}$$

In our experiments, the parameters α, β, γ, δ were set to 0.05, 0.2, 0.72, 0.03 respectively. At first, we compute each indicator for each argument separately using the above-mentioned equations (3, 5, 6, 7). Then, we combined those four indicators to calculate the argument score for discovery (11). The arguments' scores are in [0, 1]. The arguments are displayed in the arguments' list based on the arguments' score in decreasing order. As a position or an argument is made of a tree with different levels, we applied this framework recursively for each indicator for each argument from the leaves to the tree root.

H. The Argument Discovery Framework

In Section 5.7, we introduced the argument discovery scoring method. However, some users would like to focus on one or more indicators. Thus, users may need to select the argument discovery method themselves. Therefore, we are building an argument discovery framework to accommodate users' preferences. Fig. 5 depicts an overview of the argument discovery framework. The proposed framework is straightforward as follows:

1. The user selects position *P*, retrieve all *P*'s arguments.
2. For each retrieved argument *a*, calculate the indicators scores (3, 5, 6, 7) and the argument discovery method using (11).
3. List all arguments to the user by the argument discovery method score (11) in decreasing order.
4. Ask the user to build the preferred discovery method:
    a. Use the indicators score for each argument from step 1.
    b. Apply the selected discovery method based on the type of the relationship between indicators. Calculate the argument score.
5. List all arguments to the user by the selected argument discovery method score in decreasing order.

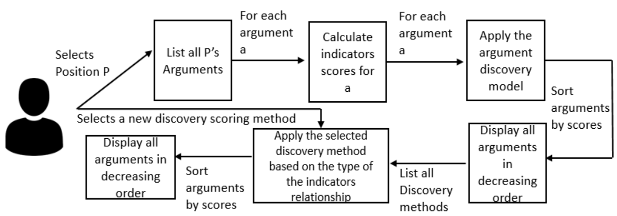

Figure 5. The argument discovery framework

VI. Empirical Dataset Collection

The dataset used in this research is from a study produced by students from an introductory level sociology class who participated in an online discussion using the ICAS for 26 days. There are four issues discussed in this study. Each issue has four predetermined positions. Therefore, ICAS has sixteen different positions discussed heavily between participants. Table I gives us more details about users' participation, arguments, and reactions for each position in ICAS.

TABLE I. Users Participation Information in ICAS

Issues	#	No. of Arguments	No. of reactions	No. of distinct users
Issue1	P0	1017	726	289
	P1	576	330	263
	P2	705	409	279
	P3	591	347	259
Issue2	P4	782	259	269
	P5	593	203	252
	P6	533	193	249
	P7	620	229	253
Issue3	P8	883	296	274
	P9	593	238	255
	P10	581	243	257
	P11	636	226	253
Issue4	P12	747	202	258
	P13	623	215	252
	P14	547	151	244
	P15	556	157	238

VII. Application Example

In this section, an application example with different scenarios is presented to illustrate the proposed framework. In this example, different argument discovery requests are used to search and recommend the most relevant arguments based on a user's preferences. Each example demonstrates a scenario from the proposed framework. All the data demonstrated in this example are from Position P0. Dataset is available upon request.

A. Scenario 1

A user submits an argument discovery request of two cooperative indicators with the recommended operator. For example, if the request is "Engagement $\oplus$ Controversy", the top 5 relevant arguments are:

TABLE II. Discovering Arguments with Cooperative Indicators

No.	Author Name	Created Date	No. Reactions	No. Arguments
a30	user53	1	0	11
a39	user106	2	0	9
a369	user58	9	0	9
a785	user303	23	0	8
a161	user87	4	2	9

We can see that all arguments discovered by this discovery method are engaging but not necessarily recent. Arguments in the above Table II had scored above the 90^{th} percentile in terms of controversy. However, the user did not specify the recentness in the request.

B. Scenario 2

A user submits an argument discovery request of two mutually exclusive indicators with the recommended operator. For example, if the request is "Recentness $\odot$ Controversy", the top 5 relevant arguments are shown in Table III:

TABLE III. Discovering Arguments with Mutually Exclusive Indicators

No.	Author Name	Created Date	No. Reactions	No. Arguments
a943	user226	25	2	4
a947	user29	25	2	3
a813	user249	24	0	4
a938	user316	25	0	2
a922	user326	25	2	1

All arguments discovered by this discovery method have scored above the 90^{th} percentile in terms of controversy. However, those arguments are the newest. a813 was created on day 24 and the study lasted for 26 days.

Because these indicators are mutually exclusive, it is hard to find arguments that scored the highest for all indicators at the same time.

C. Scenario 3

A user submits an argument discovery request of two conflicting indicators without the recommended operator. For example, if the request is "Controversy $\otimes$ Recentness", the top 5 relevant arguments are as in Table IV.

Arguments discovered by this discovery method are new, but there are some newer arguments that have not been recommended. In terms of controversy, only two arguments have scored above the 90^{th} percentile. Since those indicators are mutually exclusive, we see that the order of the arguments is different from Scenario 2. Therefore, using the not recommended operator between the indicators lead to different results or no results in some cases.

TABLE IV. Discovering Arguments with the NoT Recommended Operator between Indicators

No.	Author Name	Created Date	No. Reactions	No. Arguments
a947	user29	25	2	3
a943	user226	25	2	4
a818	user249	24	0	2
a936	user19	25	0	2
a813	user249	24	0	4

D. Scenario 4

In this scenario, we are presenting the results from applying the argument discovery model (11). The top 5 relevant arguments are:

TABLE V. Discovering Arguments by the Argument Discovery Model

No.	Author Name	Last Activity Day	No. Reactions	No. Arguments
a943	user226	25	2	4
a947	user29	25	2	3
a818	user249	24	0	2
a813	user249	24	0	4
a936	user19	25	0	2

Arguments in the above Table V are engaging and recent. Moreover, some of those arguments had scored above the 90th percentile in terms of all the argument indicators. This model does not require users to understand the argument indicators nor the mechanism of different argument discovery methods. Furthermore, this mechanism does not require additional processing since it is automatically applied after a user selects a position.

VIII. Conclusion and Future Work

In this paper, a novel framework is proposed for discovering constructive opinions in cyber-argumentation platforms. This framework discovers opinions based on four non-textual features. It identifies and measures the degree of recentness, engagement, controversy, and author influence for each opinion. Then, it combines those indicators for discovering the constructive opinions and recommends them to users. Furthermore, it allows users to select and build their own argument discovery method.

Unlike other cyber-argumentation platforms, arguments promoted by our framework are meaningful because they are discovered using a list of non-textual features while paying equal attention to each of the four essential features of opinions. Moreover, it provides users with chances to specify their preferences. Thus, it encourages constructive discourse between users that improves the quality of the discussion. Our framework has many potential applications in the context of opinions discovery. It can be adapted and customized by many UGC applications such as promoting posts on social media, reviews in online retailers and replies in online news platforms.

This work focused on discovering opinions on the position and the argument level. However, we did not perform the discovery method on the issue level. Issues may have different settings and measurements on the issue level. Alternatively, issues may exhibit other features set for issues discovery. This is left for future work.

Conflict of Interest

The authors have no conflicts of interest to declare.

Author Contributions

N. Althuniyan developed the theoretical formalism, performed the analytic calculations, performed the numerical simulations and contributed to the final version of the manuscript. X. Liu supervised the research. J. Sirrianni has developed the ICAS platform. Finally; D. Adam, N. Althuniyan and J. Sirrianni collected the data.

References

[1] R. Arvapally and X. F. Liu, "Analyzing credibility of arguments in a web-based intelligent argumentation system for collective decision support based on K-means clustering algorithm," *Knowledge Management Research & Practice*, vol. 10, no. 4, pp. 326-241, December 2012.

[2] M. Klein, "The CATALYST deliberation analytics server," *Ssrn Electronic Journal*, 2015.

[3] J. Sirrianni, X. Liu, and D. Adams, "Quantitative modeling of polarization in online intelligent argumentation and deliberation for capturing collective intelligence," in *Proc. IEEE International Conference on Cognitive Computing*, July 2018, pp. 57-64.

[4] M. M. Rahman, J. W. Sirrianni, X. F. Liu, and D. Adams, "Predicting opinions across multiple issues in large scale cyber argumentation using collaborative filtering and viewpoint correlation (In press)," in *Proc. the Ninth International Conference on Social Media Technologies, Communication, and Informatics*, 2019.

[5] Kialo homepage. [Online]. Available: www.Kialo.com

[6] Debate homepage. [Online]. Available: www.org

[7] N. Diakopoulos and M. Naaman, "Towards quality discourse in online news comments," in *Proc. the ACM Conference on Computer Supported Cooperative Work*, March 2011, pp. 133-142.

[8] J. Sirrianni, X. F. Liu, M. M. Rahman, and D. Adams, "An opinion diversity enhanced social connection recommendation re-ranking method based on opinion distance in cyber argumentation with social networking," in *Proc. IEEE International Conference on Cognitive Computing*, Milan, Italy, 2019.

[9] P. M. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games," *Artificial Intelligence*, vol. 77, no. 2, pp. 321-357, September 1995.

[10] W. Kunz and H. W. J. Rittel, *Issues as Elements of Information Systems*, Berkeley, California: Institute of Urban and Regional Development, University of California, July 1970, vol. 131.

[11] N. Althuniyan, J. W. Sirrianni, and M. M. Rahman, "Design of mobile service of intelligent large-scale cyber argumentation for analysis and prediction of collective opinions," in *Proc. International Conference on AI and Mobile Services*, June 2019, pp. 135-149.

[12] Q. V. Nguyen, C. T. Duong, T. T. Nguyen, M. Weidlich, K. Aberer, H. Yin, and X. Zhou, "Argument discovery via crowdsourcing," *The VLDB Journal*, vol. 26, no. 4, pp. 511-535, August 2017.

[13] G. Mishne, "Multiple ranking strategies for opinion retrieval in blogs," *Online Proceedings of TREC*, 2006.

[14] L. Amgoud and J. Ben-Naim, "Ranking-based semantics for argumentation frameworks," in *Proc. International Conference on Scalable Uncertainty Management*, September 2013, pp. 134-147.

[15] E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet, "Argumentation ranking semantics based on propagation," *COMMA*, vol. 16, pp. 139-150, 2016.

[16] M. Eirinaki, S. Pisal, and J. Singh, "Feature-based opinion mining and ranking," *Journal of Computer and System Sciences*, vol. 78, no. 4, pp. 1175-1184, 2012.

[17] F. Pu, J. Luo, Y. Zhang, and G. Luo, "Argument ranking with categoriser function," in *Proc. International Conference on Knowledge Science, Engineering and Management*, October 2014, pp. 290-301.

[18] Facebook homepage. [Online]. Available: www.Facebook.com

[19] Twitter homepage. [Online]. Available: www.Twitter.com

[20] H. L. O'Brien, "Exploring user engagement in online news interactions," *Proceedings of the American Society for Information Science and Technology*, vol. 48, no. 1, pp. 1-10, 2011.

[21] S. Dori-Hacohen, "Controversy Analysis and Detection," doctoral dissertations, University of Massachusetts Amherst, vol. 1084, 2017.

[22] K. Garimella, G. D. F. Morales, A. Gionis, and M. Mathioudakis, "Quantifying controversy on social media," *ACM Transactions on Social Computing*, vol. 1, no. 1, p. 3, 2018.

[23] W. Xing and A. Ghorbani, "Weighted pagerank algorithm," in *Proc. the Second Annual Conference on Communication Networks and Services Research*, May 2004, pp. 305-314.

[24] Y. Chang, X. Wang, Q. Mei, and Y. Liu, "Towards twitter context summarization with user influence models," in *Proc. the Sixth ACM International Conference on Web Search and Data Mining*, February 2013, pp. 527-536.

[25] X. Tang and C. C. Yang, "Identifing influential users in an online healthcare social network," in *Proc. IEEE International Conference on Intelligence and Security Informatics*, May 2010, pp. 43-48.

[26] J. D. Evans, *Straightforward Statistics for the Behavioral Sciences*, Pacific Grove, CA: Brooks/Cole Publishing, 1996.

[27] X. Liu, K. K. Fletcher, and M. Tang, "Service selection based on personalized preference and trade-offs among QoS factors and price," in *Proc. IEEE First International Conference on Services Economics*, June 2012, pp. 32-39.
[28] K. A. Bollen and S. J. Long, *Testing Structural Equation Models*, SAGE Focus Edition, 1993, vol. 154.

Najla Althuniyan has received her B.S from King Saud University, SA, and her MCS from California State University, Fullerton, USA. She worked as a senior developer for four years and as a lecturer in Prince Sultan University, SA for three years. Currently she is pursuing her Ph.D. degree at the University of Arkansas. Her research area is cyber-argumentation based analysis.

Xiaoqing (Frank) Liu is currently a professor and department head and holds the Rodger S. Kline endowed leadership chair in the Department of Computer Science and Computer Engineering, at the University of Arkansas in Fayetteville, USA. His current interests include software engineering, service computing, data analytics based recommendation systems, cyber argumentation based social media, cyber manufacturing, motion tracking, and applied artificial intelligence. He served as a PI, Co-PI, or faculty participant of 29 funded research projects and published more than 150 referred papers in numerous journals and conferences, such as TWeb, TSC, JVR, SPIP, SQJ, ICSE, JSS, AAAI, ICWS, and CSCW. He received his PhD in computer science from the Texas A&M University in College Station in 1995.

Joseph W Sirrianni is currently a Ph.D. student in the Department of Computer Science and Computer Engineering, at the University of Arkansas in Fayetteville, USA. His research area centers on cyber argumentation based social media. He received his bachelor of science degree from the University of Arkansas in Fayetteville in 2016.

Douglas James Adams received his BA in Sociology in 1982 from Augsburg College in Minneapolis, Minnesota. He earned his MA in Sociology in 1991, and his Ph.D. in Sociology in 1996 from the University of Arizona, Tucson. Douglas Adams, Ph.D. has been a Faculty Member of the Department of Sociology and Criminal Justice at the University of Arkansas (UA) since 1995. He has served as Chair of the University's Institutional Review Board (IRB) for the last five years, and as a board member for the last eight years. In addition, Dr. Adams has served on the Academic Advisory Board of the Mack-Blackwell Rural Transportation Center since 2006. He has published many research papers in the field of Criminology, Sociology.