Text-Based Clustering and Analysis of Intelligent Argumentation Data

Eric Barnes
Department of Computer Science
Missouri University of Science and Technology
Rolla, United States
ebyt7@mst.edu

Xiaoqing (Frank) Liu
Department of Computer Science
Missouri University of Science and Technology
Rolla, United States
fliu@mst.edu

Abstract—Argumentation is a method by which stakeholders exchange their viewpoints and rationale in the form of arguments in an organized manner in order to conduct collaborative decision making. Many online systems have been implemented in order to provide geographically distributed stakeholders with a structured method of argumentation. However, as these systems collect large amounts of arguments; it can be difficult to readily assess the major concerns which drive the discussion. This work presents a method for clustering and classifying a set of arguments, collected through an online argumentation tool, in order to model major concerns in an argumentation process. These clusters are further analyzed to provide a qualitative understanding of the influence they have on the decision making process.

Index Terms—Argumentation, Collaborative Decision Making, Text Clustering,

I. INTRODUCTION

In the Software Engineering process, there are often many points at which a selection must be made from a number of mutually exclusive alternatives. If this decision is to be made collaboratively, each stakeholder will have their own idea of which alternative is ideal with respect to their own needs and experience, and will disagree with one another about which alternative to take. When decision making proceeds without an understanding of stakeholder conflict, decisions can be made which do not adequately address the concerns of this group.

The process of argumentation in part addresses this by providing stakeholders with an organized platform for exchanging their knowledge and the rationale behind their preferences in the form of arguments. With the structured representation of stakeholder rationale, decision makers are better equipped to select alternatives which represent the collective needs and expertise of the group. Online argumentation tools further enhance this process by allowing geographically distributed stakeholders to exchange arguments asynchronously, with such arguments collected and recorded in a logical structure. Such structures reflect how arguments relate to one another and provide context for each argument made. This can be reviewed over the course of the decision making process in order to analyze the rationale behind each viewpoint, and attempt to determine the overall consensus before making a selection.

However, as the number of arguments increases, it becomes increasingly difficult to maintain an overall understanding of the argument in progress, as each argument must be read individually for their content to be understood. Many of these arguments will refer to similar subjects or concerns. Concerns which manifest in a large amount of arguments represent influential topics of conversation and debate. If a stakeholder had some idea of what these major concerns were, they would have a quick access to the broad themes of the arguments in progress. However, arguments which relate to the same subject are not necessarily organized as such in the argument structure, as references particular concern can arise in response to any number of threads of debate. This is particularly the true when the number of arguments is large, as stakeholders themselves lose sight of the overall discussion.

It is not practical to demand that participants form logical argument structures which are perfectly organized according to strict topics. They should be free to use all relevant information and voice their concerns in the course of debate. Because computers have no issues parsing large numbers of arguments, it is more reasonable for the argumentation system itself to perform analysis on logically distributed arguments to determine what the major concepts involved in discussion are, organize arguments according to these concepts when applicable, and present this information to participants.

This work proposes the use of text clustering techniques to group arguments according to the concerns they represent. This provides stakeholders with an view of which topics are most strongly represented in the form of arguments. The system possesses an inference engine which can calculate the degree of support held by the available positions or alternatives by relating the weights of their direct and indirect responding arguments.

Once text clusters are formed to describe major argument concerns, additional analysis can be performed using the logical argument structure and the weights of each argument to determine what effect the clustered arguments have on discussion.
II. RELATED WORKS

A. Argumentation Systems

Philosopher Stephen Toulmin proposed a model of argumentation which has formed the basis for a wide scope of computer-based argumentation systems [1]. The first such method was gIBIS, which displayed arguments, issues, and positions in a unified graph [2]. HERMES is a decision support tool which describes arguments and evidences as nodes in a logical hierarchy [3]. Evidences in this system are specific arguments which attach themselves to conflicting argument in order to provide one with more concrete proof. The system then can use the weights of all arguments to find the most favored alternative. Chen-Junn Huang developed an argumentation system which analyses arguments, assesses their quality, and sends feedback to users depending on their calculated skill level [4]. Collaboratorium [5] is another argumentation system which allows stakeholders to exchange rationale and ideas by posting them to a visual representation called the argument map.

In all of these systems, all arguments must be read individually in order to parse their contents. This is feasible for following individual threads of discussion, but it is not practical to attempt to read an entire argumentation structure in order to gain an overall understanding of stakeholder concern. This work is unique in that it uses text clustering to group arguments according to important discussion topics, and then using the argumentation structure to provide additional context to these argument groups.

III. OVERVIEW OF TECHNIQUE

This process aims to provide both an overview of the important topics which arise during an argumentation process. Important in this sense refers to concepts which appear in a relatively large number of arguments. Arguments are collected via our online intelligent argumentation system including both their text component, weight, and logical position in the argument hierarchy.

These arguments are then clustered and classified based on their text components, the labels for these clusters denote frequently referenced concepts and terms, with labels describing the largest clusters denoting the more important terms. Text clustering is carried out using Lingo algorithm [6], a process developed for clustering search web search results based on the textual content of their snippets. Lingo was chosen because both arguments and snippets are relatively small compared to other types of documents. With the information provided by the argument clusters, and the analysis thereof, stakeholders and decision makers can have a better understanding of the broad topics of argumentation at a glance.

IV. ONLINE INTELLIGENT ARGUMENTATION SYSTEM

An online argumentation system has been implemented [1] which organizes arguments in a logical tree structure, and also allows stakeholders to attach weights to their arguments, denoting the degree of support or attack they are responding with. The argumentation tree consists of the following elements:

- **Issue** - This is a design problem or complication that a particular project faces. The solution to this issue is to be found through collaborative decision making supported by the intelligent argumentation system. A project may have any number of potential issues, and the scope of such issues can include any decision that must be made within the software development process.

- **Position** – This is a potential solution to a given issue. In the context of software engineering it could be a component to add to the system, a design decision, or any other business decision problem which needs to be resolved for the project.

- **Argument** - This is a concise statement partially giving the rationale behind a stakeholder’s preference. Arguments attached to positions contain a rationale which partially explains a stakeholders support or lack of support for that position. Arguments attached to other arguments attempt to reinforce or contradict the statement made by the other participant with additional rationale. An argument is either supporting or attacking depending on its given weight: a number ranging from -1 to 1. Negative weights indicate attack, while positive weights indicate support. The magnitude of the weight reflects the degree of support or attack.

These elements compose a structure which will be referred to as the argument tree. A representation of a sample argument tree can be seen in Figure 1. In this figure, squares correspond to nodes in the tree, and edges indicate a parent/child relationship. In Figure 1, Argument 3 is a direct response to

![Figure 1. Sample Argumentation Tree](image-url)
Argument 1, but is also an indirect response to Position 2 at its root. The argumentation framework is capable of inferring the indirect relationship of all arguments to the corresponding position at their root. The normalized aggregate of all reduced weights is defined as the favorability factor of that position. This process is described fully in previous work.

V. OPERATION OF LINGO CLUSTERING

While the complete workings of Lingo are outside the bounds of this work, a brief overview of the algorithm is given as the following four steps:

Pre-processing: Text inputs are pre-processed in the following ways. Stemming is carried out to find the semantic representation of inflected words. Stop words are ignored to remove commonly used words which have no meaning on their own, such as conjunctions, articles, etc. Finally, text is divided into words and sentences through the use of text-segmentation heuristics.

Phrase extraction: A modified version of SHOC’s phrase discovery algorithm is used to extract potential cluster labels from the input documents. To be a potential label, a phrase must satisfy a set of prescribed conditions such as frequency of use, etc.

Cluster-label induction: Matrix decomposition is used to summarize cluster labels via singular value decomposition. The relevance of different potential labels are compared for each document, and the most relevant labels for the group are selected as cluster labels.

Cluster-content allocation: Queries are performed on each document for each cluster label, if the similarity between the document and the label exceeds a specified threshold, the document is added to the cluster.

VI. ANALYSIS OF ARGUMENT CLUSTERS

Because the logical placement and weight of arguments in the argumentation tree is integral to their meaning, additional analysis can be performed based on this in order to properly understand concern clusters.

A. Cluster-specific Favorability Factor

In the online argumentation system, the weight of each argument can be until it relates directly to some position. As the overall favorability of each position is calculated as the normalized summation of all reduced arguments responding to it, cluster specific favorability can be similarly derived by only considering arguments from a particular cluster.

Just as the favorability factor represents which alternatives are more strongly supported by all arguments, cluster-specific favorability factors represent the degree to which arguments pertaining to each concern support or attack the given alternatives. This can provide a more nuanced view of favorability, describing not just overall support, but relating that degree of support to specific topics, giving a potential for exploring the strengths and weakness of each position.

B. Unique contributors

While the cluster size represents the total number of arguments in a cluster, it may be useful to know how many stakeholders have expressed an argument relating to each concern. For instance, a cluster may contain a large number of arguments, but these arguments are only posted by a small portion of the stakeholder population. Analysis of a cluster amount of unique contributors can be used to help determine how widespread a particular concern is among stakeholders. Because each participant includes their unique id as part of their argument, it is relatively simple to look at an argument cluster and count the number of unique contributors.

C. Response Metrics

Arguments which have a large amount of responding arguments, both direct and indirect, have a high amount of influence on the argumentation process, as their very existence provokes additional debate and discussion. Due to the structure of the argumentation tree, the direct and indirect responses to an argument can be found by isolating a sub-tree in the argumentation system which is rooted at the argument in question. The calculation of the reduced weight for all of these sub-arguments must include the weight of the influential node. Additionally, the aggregate weights of these responses are also vital information, as an argument which receives a strongly positive response indicates an opinion shared by multiple stakeholders, while a strongly negative one indicates otherwise. The process for quantifying an individual arguments Reaction weight, is identical to finding the favorability of a position: take the normalized sum of all responses reduced to the level of a direct response.

This principle also applies to argument clusters. Their influence is not limited to their own aggregate favorability factors and size, the level and nature of the response that the cluster provokes should also be considered. This is expressed through two additional metrics. The response size and cluster reaction are both found by tracing each argument which is not already in the cluster upwards until either a position or argument in the system is encountered. If the trace encounters an argument within the cluster, then it is a response and the response size is incremented. Additionally, the weight of the response is reduced until it is directly responding to the in-cluster argument, and this amount is added to the cluster reaction. The total reaction for a cluster is taken as the summation of all reduced responding argument weights normalized with respect to the total response size, giving stakeholders an idea of scope and nature of the response provoked by a cluster.

VII. EMPIRICAL STUDY

An empirical study was carried out to simulate the use of the argumentation system for software engineering decision making. 24 participants were to take the role of stakeholders in a medium scale software development firm which had to decide on a metrics system to employ: no metrics system, represented as position 1, a lightweight metrics program represented as position 2, and a comprehensive metrics program, represented as position 3. 314 arguments in total were collected. Argument clustering was performed via Lingo with a base cluster count of 15, resulting in 26 total clusters, with 107 arguments listed as “Other Topics”.

424
While the base cluster count could be increased to group additional arguments, it was kept at 15 to focus on the more influential clusters and obtain meaningful labels. The purpose of this method is not to necessarily categorize all arguments, but to locate and analyze the important topics of discussion. The top-10 most heavily populated clusters are given in Table 1, with the first column giving the cluster label as derived by lingo, column 2 giving the size of the cluster, columns 2,3,4 giving the cluster specific favorability factors in response to P1, P2, and P3, column 5 giving the total number of responses to the cluster, column 6 giving the aggregate reaction weight, and column 7 giving the number of contributors to the cluster.

### Table 1: Analysis of 10 largest argument clusters

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Products</td>
<td>31</td>
<td>-0.28</td>
<td>0.61</td>
<td>-0.06</td>
<td>10</td>
<td>-0.25</td>
<td>14</td>
</tr>
<tr>
<td>Products Developed</td>
<td>30</td>
<td>-0.42</td>
<td>0.65</td>
<td>-0.20</td>
<td>9</td>
<td>-0.16</td>
<td>12</td>
</tr>
<tr>
<td>Metrics can Measure</td>
<td>29</td>
<td>-0.25</td>
<td>0.39</td>
<td>-0.18</td>
<td>10</td>
<td>-0.31</td>
<td>12</td>
</tr>
<tr>
<td>New Technologies</td>
<td>28</td>
<td>-0.30</td>
<td>0.49</td>
<td>-0.28</td>
<td>11</td>
<td>-0.03</td>
<td>11</td>
</tr>
<tr>
<td>Time Money</td>
<td>27</td>
<td>-0.80</td>
<td>0.81</td>
<td>-0.85</td>
<td>16</td>
<td>-0.14</td>
<td>8</td>
</tr>
<tr>
<td>Reducing Requirements</td>
<td>22</td>
<td>-0.85</td>
<td>0.52</td>
<td>-0.40</td>
<td>16</td>
<td>0.15</td>
<td>9</td>
</tr>
<tr>
<td>Opposing Argument</td>
<td>18</td>
<td>-0.107</td>
<td>0.51</td>
<td>-0.59</td>
<td>3</td>
<td>-0.1</td>
<td>2</td>
</tr>
<tr>
<td>Revenue Used for Quality Assurance</td>
<td>17</td>
<td>-0.304</td>
<td>0.42</td>
<td>0.06</td>
<td>5</td>
<td>-0.12</td>
<td>7</td>
</tr>
<tr>
<td>Loss in Revenue</td>
<td>16</td>
<td>-0.49</td>
<td>0.56</td>
<td>-1.0</td>
<td>11</td>
<td>-0.07</td>
<td>10</td>
</tr>
<tr>
<td>Measure Productivity</td>
<td>16</td>
<td>-0.12</td>
<td>0.78</td>
<td>0.0</td>
<td>2</td>
<td>-0.09</td>
<td>8</td>
</tr>
</tbody>
</table>

**A. Qualitative Interpretation of Results**

1) **Labels**

While the top two clusters are the most general, both relating to Software product development, subsequent clusters provide more interesting information, particularly that clusters are concerned with topics such as New Technologies, Time, Money, Reduction of Requirements etc. Each cluster could be browsed to examine their associated arguments directly for a better understanding of their meaning. For instance the Reduction of Requirements arguments were generally concerned with the business habit of reducing requirements to meet deadlines.

2) **Cluster Specific Favorability Factors**

All clusters collected favored position 2 over both 1 and 3, as is consistent with the overall favorability factor calculated by the system. However the degree of this favorability varies somewhat from cluster to cluster. For instance, the Time Money cluster came down particularly harshly on both P1 and P3 and was the most supportive of P2. Conversely, the Revenue used for QA cluster was slightly supportive of P3 and only moderately supportive of P2.

**C. Influence Analysis through Contributors and Response**

For most clusters, the collective reaction was slightly to moderately negative. One exception was Reducing Requirements, which was slightly supported by 16 responses, and was tied with Time Money for the largest amount of responses. Although it was not the most heavily populated cluster in terms of arguments total or contributors, the reaction it provokes makes it a particularly influential cluster.

Conversely, Opposing Argument, despite its high argument count, only has 2 contributors, and a negligible degree of response. Interpreting these metrics, it is clear that this is not an influential concept in relation to the others.

**VIII. CONCLUSIONS AND FUTURE WORK**

Navigating a logical argumentation structure becomes increasingly difficult as the number of arguments increases, making the maintenance of the overall discussion challenging. By clustering arguments based on their textual component, it is possible to identify concerns according to clusters formed by these common terms and themes. Furthermore, due to the weight and logical organization of arguments, it is possible to perform additional analysis to determine how these concerns affect the overall argumentation process. Future work is planned to perform analysis on the argument cluster contributors to determine if it is possible to group stakeholders in terms of their frequently expressed concerns, and analyze the relationships among contributor groups.

**REFERENCES**


