Rethinking Online Discussions

Inaugural-Dissertation

zur Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Christian Meter

geboren in

Remscheid

Düsseldorf, Juni 2020

aus dem Institut für Informatik
der Heinrich-Heine-Universität Düsseldorf

Gedruckt mit der Genehmigung der
Mathematisch-Naturwissenschaftlichen Fakultät der
Heinrich-Heine-Universität Düsseldorf

Berichterstattende:

1. Prof. Martin Mauve

2. Prof. Stefan Conrad

Tag der mündlichen Prüfung: 05.06.2020

To my wonderful girlfriend, my family and my friends

Abstract

A rethinking of how discussions are supported on the Internet is necessary. We have reached a point where many people would like to exchange ideas via the Internet. In doing so, they are making too many text contributions to be evaluated or read by other participants, due to their sheer mass.

Much of this can be traced back to structural problems in online discussions, which do not allow a productive exchange with many discussants. Often, a lightweight overview of the discussion, real scaling abilities or concrete interaction possibilities are missing. This leads to a lack of ability to respond in a concrete and productive way to arguments brought by other participants.

Motivated by these problems, this dissertation focuses on three topics that can improve and support current online discussions. First, we work on a new way to guide the discussants through the discussion and capture arguments in a structured way. We call this approach "dialog-based discussion". In dialog-based discussion, participants are guided through a natural language dialog and can specifically address the individual components of the other participants' arguments to explicitly attack and invalidate or support these components.

Second, we consider integrations of this new type of discussion in order to change argumentation styles in as many areas as possible. In doing so, we present concrete embeddings of dialog-based discussions for web pages, which can enhance the user-experience on handling increased numbers of argumentation contributions. This technique enables the user to address specific points in the article and incorporate them into their own argumentation.

Third and finally, we concentrate on the reductions of redundant contributions by presenting a distribution structure of arguments in which, for example, contributions from the discussions of various online editorial offices can be shared amongst each other. These arguments are then presented to the respective readership in such a way that they can be reused. In this way we shorten discussions on the same topic, since it is possible to import previous arguments, so that nothing "old", i.e., arguments that have already been discussed, needs to be added.

All projects are freely available in the form of open-source software. We tested our software scientifically with participants untrained in argumentation theory in order to determine their practical suitability. Feedback from the users in the field test is then used to refine our projects. Comparative tests were also conducted using our software and established commentary systems to evaluate our idea of dialog-based discussion. As a result, we produce academic software projects that can now be developed to market maturity for widespread use as well as being used in academia in their current form.

Zusammenfassung

Ein Überdenken der Art und Weise, wie Diskussionen im Internet unterstützt werden, ist notwendig. Wir sind an einem Punkt angekommen, an dem viele Menschen sich über das Internet austauschen und dabei eine hohe Anzahl an Textbeiträgen liefern. Das hat zur Folge, dass diese Beiträge aufgrund ihrer hohen Masse kaum ausgewertet oder von anderen Teilnehmenden gelesen werden können.

Vieles lässt sich auf strukturelle Probleme von Online Diskussionen zurückführen, die einen produktiven Austausch mit vielen Diskutierenden gar nicht erst ermöglichen. Denn häufig fehlen leichtgewichtige Übersichten, echte Skalierung oder konkrete Interaktionsmöglichkeiten, um direkt auf das vorher Gesagte der Teilnehmenden eingehen zu können.

Davon motiviert werden in dieser Dissertation drei Themengebiete behandelt: Zunächst haben wir an einem neuartigen Ansatz gearbeitet die Diskutierenden durch die Diskussion zu leiten und dabei Argumente strukturiert zu erfassen. Wir nennen diesen Ansatz „dialogbasiertes Diskutieren“. Beim dialogbasierten Diskutieren werden Teilnehmende durch einen natürlichsprachlichen Dialog geführt und können auf die einzelnen Bestandteile der Argumente anderer Teilnehmenden konkret eingehen.

Danach betrachten wir Integrationsmöglichkeiten dieser neuen Diskussionsart, um die Diskussionskultur in möglichst vielen Bereichen zu verändern. Dabei stellen wir konkret Einbettungen in Webseiten vor, da gerade dort, wo kontroverse Artikel veröffentlicht werden, ein Anstieg von Argumentationsbeiträgen zu beobachten ist. Ein weiterer Vorteil dieser Strategie ist, dass es damit möglich ist auf konkrete Stellen im Artikel einzugehen und sie in die eigene Argumentation mit einfließen zu lassen.

Abschließend befassen wir uns mit der Reduktion redundanter Beiträge, indem wir eine Verteilstruktur von Argumenten präsentieren, bei der Beiträge aus den Diskussionen verschiedener Online Redaktionen unter den teilnehmenden Redaktionen geteilt und der entsprechenden Leserschaft so präsentiert werden, dass diese wiederverwendet werden können. Dadurch reduzieren wir redundante Beiträge und können gleichzeitig Diskussionen zur selben Thematik auf ein produktives und übersichtliches Maß verkürzen.

Alle Projekte sind wir in Form von Open-Source Software nutzbar und wissenschaftlich mit, in Argumentationstheorie untrainierten, Teilnehmenden getestet, um die Praxistauglichkeit festzustellen. Mit den Rückmeldungen der Probanden wurden die Softwareprojekte anschließend verbessert. Um unsere Idee des dialogbasierten Diskutierens kritisch zu hinterfragen, führen wir auch Vergleichstests zwischen unserer Software und etablierten Kommentarsystemen durch. Dadurch gewinnen wir wertvolle Erfahrungen, die dann wiederum in die Entwicklung der Projekte einfließen können. Entstanden sind Softwareprojekte, die nun sowohl für einen großflächigen Einsatz zur Marktreife weiterentwickelt, als auch in der aktuellen Form weiter in akademischen Projekten genutzt werden können.

Acknowledgements

During this thesis there were many people supporting me and my work regarding this topic. Now, it is time to thank all of you and I hope that I do not forget anyone. If so, it is definitely not intentional by any means.

My biggest thanks goes to my supervisor Prof. Martin Mauve, who constantly supports me, brings new ideas and pushes our papers to a higher level. I learned a lot about scientific writing and it was not always easy, but all the time very helpful. Thanks to Prof. Conrad, who is available as a second reviewer for this thesis.

Big thanks goes to my colleague Alexander Schneider as well, who always shares an office with me. Our discussions can be very challenging, but all along produce productive results. I am looking forward to more controversial conversations in the future.

It is always nice to discuss and work with my colleagues. Thanks for the intense work with Tobias, jointly pushing forward the idea of dialog-based discussions. I am sure that it is not always easy to work with me. Thanks to more colleagues, namely Raphael, Andre, Daniel, Björn, and Markus. It is always a pleasure to work with you and share thoughts at lunch. Teaching students is a hard job, but working with Jens is challenging and enriching all the time. I am sure that I learned more than the students in our lectures and exercises. Furthermore, a lot of work is done by our assistant researchers Marc, Jan, and Teresa. Your work in this project will always be appreciated.

Thanks to our great and helpful secretaries and system administrators Sabine, Angela, Claudia, Thomas, and Guido — your support makes working at our chair productive and comfortable.

A special thanks goes to my girlfriend Johanna, who helped me to proofread this thesis in the first place and who had to tangle with my English skills, which are not always the yellow from the egg. Thanks for your support during these stressful times. Furthermore, Philipp and Mike helped me additionally to my colleagues to proofread this thesis. Your comments are very valuable and of great help.

Last but not least, thanks to my friends, especially to Flo, Werner, Mona, Malte, Kay, and more, who supported me throughout the past years. And thanks to the scouts bringing me closer to nature and to myself.

Contents

Chapter 1

Introduction and Motivation

Technology and computers quickly evolved in the last years and holding pace is a hard job. The rise of artificial intelligences seems to be unstoppable and a rapid development occurs in the technologies we use in our daily lives. But not every technology evolved in this rapid pace. Since the early days of the Internet, the evolution of discussions on the Internet was quite poor which is why the way people are discussing nowadays has hardly changed. This is especially problematic, because using the Internet seems quite natural to get a big crowd of people involved into participation processes or to solely get to know the crowd's opinion about specific topics.

For discussions on the Internet, lists, comment sections or forum systems are mostly being used. Because of their simplicity, these techniques enjoy quite a lot of popularity and are the current de facto standard. These structures share the substantial problem that they generate masses of unstructured text entries, which are currently not automatically analyzable, at least not satisfactory. Especially hot topics, like Climate Change or the Brexit, provoke many users to join a discussion. By this, it is not uncommon to generate, for example, 4,000 – 8,000 comments in a few hours for a single article or post (Bundesregierung via Facebook 2019; Guardian News & Media Limited 2019). These comments need then to be reviewed and moderated. Content creators, like the *Federal Government of Germany* or *The Guardian*, have a special interest in the contributions of their readers, because understanding what the visitors are interested in, enriches journalism and increases understanding. The amount of comments and discussion-contributions is often very high. For example, The Guardian analyzed 70 million comments from their platform, which were collected from 2006 to 2016 (Guardian News & Media Limited 2016). All of these comments are reviewed by a team of moderators blocking around 2% of these comments because of violations of their community guidelines. This results in a huge amount of work for the sole reason of collecting the reader's opinion. These comments are full of redundant information, because nearly every participant of the discussion does not read all the previous comments.

In other areas of application, like online participation processes in the political context, our ideas could lead to significant improvements in the discussion-culture. Oftentimes, municipalities or cities want to involve their citizens into political decision processes or the public itself wants to be involved in law-making processes (Alsina and Martí 2017). Using the Internet to reach every interested person, seems quite natural. Yet, the systems used to collect the citizen's opinions, are the ones described above and have the same inherent flaws in this context; namely too many text comments, which need to be processed manually.

The major problem of discussing is the unstructured nature of text entries. To solve this problem, researchers developed the first discussion systems, which provided skilled and trained users an interface to add their arguments into an argumentation graph (see Section 2.3 for more information). It needs to be considered that these systems are usually developed for professional users and are therefore not easily understandable by untrained users, i.e. participants without a background in argumentation theory.

This is why we need to approach this problem from a different angle: put the *people* into the center and try to understand the problem from the point of view of common users, not from those with a scientific background. Our research focuses on the field of Online Argumentation with these participants in the center. Using this unique approach, we want to build software projects, which follow the way of natural discussions. These projects need to be intuitively usable, in order to require no training or specialized knowledge from those taking part in an online-discussion. As a result, we want to produce discussion data, which is applicable for automatic and algorithmic processes. Therefore, our main contribution is to introduce *structure* into unstructured text entries as we experience it in current discussions on the Internet (see Chapter 2).

Another suggested solution processing the masses of text entries comes from the research field of *argument mining*, where computer programs try to extract premises, conclusions and relations from user-provided textural corpora (Liebeck et al. 2016). Using this idea, structured data can be extracted, to be thereafter used for further processing (Lippi and Torroni 2016). Even though, extracting arguments from free texts is still a difficult task, which can still be very challenging for trained people annotating the arguments (we describe this more in Meter et al. 2020). However, argument mining is a promising research field, which will be very important in the future, but for now the accuracy is oftentimes not ideal (Budzynska et al. 2014; Lippi and Torroni 2016) and arguments are overlooked by the algorithms.

Oftentimes, argument mining comes into play *after* the arguments are produced by the users, which is why argument mining needs to *interpret* the user's input. We, however, *explicitly* ask the participants for the premises and conclusions and infer the argument's relation by the current context in the discussion. More on this follows in the next section.

1.1 Problem Statement and Research Questions

This section concentrates on the problems we are experiencing when it comes to discussions on the Internet and the research questions, which can be derived from these problems. The main aspect this work focuses on is to understand the way people want to interact or discuss in real discussions and using the gained knowledge to develop software tools. This approach of thinking places the participants of discussions in the center of the development process. By this, we gain software that is usable by untrained users.

The user's arguments need to be collected in a natural and easily understandable way enabling us to construct interconnected argumentation graphs, called the *Web of Reasons* (WoR). In a WoR arguments are connected by their logical relationships. These relationships are constructed through our argumentation system by the users themselves leaving us with the benefit of structured data, where the user's conclusions and premises are correctly connected to each other. We are achieving this by performing a time-shifted dialog between our argumentation system and the participants which we are calling *Dialog-Based Online Argumentation* (DBOA) (Krauthoff 2018; Krauthoff et al. 2016, 2018). Former and joint work has been done in this field and this dissertation is driving the idea of DBOA further.

Leaving this first part behind, our focus is now on developing ideas how to embed dialog-based discussions into web contexts. These embeddings must be *lightweight* and *non-obstructive* to encourage normal users to participate in this new style of discussing.

At last, we examine how arguments can be interpreted as a resource, which can be looked up and reused in discussions. Understanding ownership of arguments and developing a way of distributing these arguments, similar to the idea of the Argument Web (Bex et al. 2013; Rahwan et al. 2007), forms the last part of this dissertation.

In the following paragraphs the derived research questions are formulated, which will be covered by the publications in the next chapters.

Missing *Structure* in Discussions Losing the overview is one of the major problems in online discussion. Current systems, like forums, comment sections or pros and cons lists, encourage the users to provide long texts containing their opinion and their arguments for or against a specific topic (see Chapter 2 for a more detailed view). These posts do not have to follow any argumentation scheme, so there is not necessarily a real connection between the arguments.

Sorting the posts by their timestamps chronologically or calculating the relevance of a post to select it as the next presented post, does not encourage to follow the discussion as a whole. The former sorting mechanism confronts users with too many posts, the latter might miss some relevant posts because of some algorithmic decisions. Therefore, masses of texts loosely coupled in a discussion about some topics are a genuine problem when it comes to constructive dialogs.

Structure in our context means that we have knowledge about the arguments and their relations so that they are analyzable and understandable. We are achieving this by using formally structured dialogs, which can be defined using a *Dialogue Game Description Language* (DGDL) (Bex et al. 2014a; Wells and Reed 2012). This formalization is being used to conduct a dialog game with the participating users, where untrained users are interacting with the software naturally.

On the one hand we are building and researching this kind of software to enhance the discussion-experience for the users. On the other hand we have the goal to automatically gain knowledge about the contents of the debate to form an argumentation graph. This leads us to the following research question: *How can we collect the user's opinions in a way that we are producing structured data?*

Applied Argumentation Theory without Training Typical users are not trained in argumentation theory, so they are not used to technical terms, e.g. rebuts, undercuts or undermines. This leads to a major problem when it comes to software projects, which should be usable by any user: designing software that is kept simple, creating no technical border that would discourage users.

Visualizing relevant attacks or supports in the context of argumentation might be considered as a solution, but visualizations are prone to different subjective interpretations. Describing argumentative reactions in text leads to an overload, which would be counterproductive to a productive exchange. Finding the happy medium of both worlds is one of the main objectives during the development.

Therefore, our software needs to be as easy as possible in the first place, so that untrained participants of the discussions are able to formulate their arguments naturally. *How can we transform argumentation theory in a way, that untrained and unskilled users are able to utilize it in their own discussions?*

Embedding Dialog-Based Discussions People are having conversations all over the Internet. Therefore, we have to spot the relevant locations, e.g. newspaper articles or blog posts, where users meet virtually. As a constraint we can define that we focus on the locations, where productive discussions are taking place or should take place first. These spots are relevant, because our targets are constructive debates as opposed to comment sections, where most users rather strive to announce their opinion, not seeking a fact-based exchange.

We have to think about slim integrations into existing systems and web applications. The integration should not disturb the users due to a massive presence on the site.

Introducing a new system to discuss online, presents a challenge. Thus thinking about a lightweight integration, which does not disturb the normal discussion flow, and integrates as easy as, e.g., comment sections, is of importance. *How can we integrate dialog-based discussions non-disturbingly, so that untrained users can follow its flow?*

Scalability Especially where hot topics are concerned many people get involved in the discussion process. This leads naturally to a greater number of arguments and sub-discussions. Current systems provide long lists of proposals and arguments, which have to be read by the users. Therefore, these systems do not scale with many users.

Thinking about the way how arguments are presented could significantly decrease the number of arguments, which have to be read prior to the participation in a discussion. The derived question reads as follows: *How can arguments be presented to the users that they need minimal knowledge about discussions to participate in them?*

Explicit Interactions with Online Articles Most comment sections do not offer the possibility to refer to explicit passages, e.g. in an online article. Usually, people add comments and when referring to a sentence, fact or argument they have to manually do so in their comments. Mullick et al. 2019 uses Machine Learning and Neuronal Networks to derive the referred position in the article. Creating the explicit possibility to directly interact with the article, is a desired feature (Mullick et al. 2019). *How can it be made possible to explicitly interact with an article?*

Joining a Discussion If users want to join a discussion on the Internet, they ideally have to read all the comments to understand the opinions and posts other users made. But in reality most users are not following the whole conversation before providing their own opinion. This leads to redundancy within the discussion and a lack of direct interaction with the arguments and ideas other participants brought before, because of missing information about the previous process.

Providing sufficient information to new users is of great significance for the whole progress of a discussion. Gaining a satisfactory overview with only a few steps necessary to directly interact and react with arguments could improve the quality of a discussion, which means in this case that redundant data is being reduced and the argumentation itself is more coherent.

We focus on this problem and provide a solution, allowing side-entries into the discussion without having to know all previous arguments in the (sub-) discussion. This is possible because of the structure we gain with our software tools, so that a user only may focus on the relevant arguments and can skip other, currently unimportant, ones. *How can new users quickly join a discussion via side-entries?*

Arguments as a Resource Discussions about similar topics are usually held all over the Internet. It seems natural that arguments from these discussions are sometimes the same, nevertheless the users have to write them once again on their own.

Having the possibility to reduce redundancy, e.g. same arguments or argumentation structures, and reusing (at least parts of) discussions could lead to more overview of the debate. Furthermore, statements that were already sufficiently discussed somewhere else, may be referenced in order to avoid debating the same topics over and over again. *How can (parts of) discussions and their corresponding arguments be reused for the sake of reducing redundant arguments?*

1.2 Contributions

This dissertation consists of three major parts that deal with the research questions explained in Section 1.1. At first, we present a novel approach to discuss on the Internet targeting the problems of current systems. Afterwards, we experiment with integrations of our software tools into different contexts and evaluate them to see how they are accepted. At last, we look on the structure of discussions and propose solutions for treating arguments as a resource to make them reusable and importable into foreign discussions.

We present six peer-reviewed papers, one peer-reviewed demo and findings from more field experiments, which are currently under review.

1.2.1 Dialog-Based Online Argumentation

The initial research of this project is based on the work done in Krauthoff et al. 2016. I joined the working group in 2015, we jointly worked on this topic and after he left the project our working group took over the project lead to provide a future for the project.

Chapter 3 focuses on the research being done on our dialog-based online argumentation-approach leading to our *Dialog-Based Argumentation System* (D-BAS), which is our own implementation of a *Dialogue Game Execution Platform* (DGEP) (see Bex et al. 2014a). After the first prototype was presented in Krauthoff et al. 2016, we realized that some major rework regarding the usability needs to be done, because our focus is to build software for the people. We enhance usability, reduce masses of texts by substitutions with colors and keywords, leading to a new release of D-BAS (Krauthoff et al. 2018). This release successfully implements several feedback options, which can be used by untrained users. Additionally, we introduce more democratic processes into the discussion software by developing a decentralized moderation system, enabling users with sufficient reputation to moderate (parts of) the conversation. The new version of the argumentation system is then first tested with untrained users in Krauthoff et al. 2017.

Developing a standalone-version of D-BAS forces users to use our web application if they want to discuss productively in a dialog-based fashion. It sounds smarter to us to bring dialog-based discussions to the users instead of the other way round. Therefore, we introduce an *Application Programming Interface* (API), which makes D-BAS's DGEP accessible and open to other applications, leading us to the next major part of this dissertation.

1.2.2 Integrating Dialog-Based Online Argumentation

Crucial for the success of a new way of discussing on the Internet is easy access for the people. Therefore, in Chapter 4 we are focusing on more applications of dialog-based discussions in different web-contexts. After opening the argumentation logic of D-BAS via an API, it is possible to develop new applications. The first application is discuss, which brings a lightweight embedded frontend for the Web (Meter et al. 2017a). discuss uses D-BAS in the backend to perform the argumentation logic. This enables, for example, integration into arbitrary online news articles.

Furthermore, we are working on integrations into Social Networks. As a first approach, we want to bring dialog-based discussions to Facebook because of the high number of users (2.5 billion as of Q4 2019, Statista Inc. 2020). We developed the Social Agent Jebediah, which integrates into this network, converts participating users from Facebook to D-BAS users and enables dialog-based discussions directly via Facebook's Messenger interface (Meter et al. 2018a).

Bringing argumentation-software to the people is one of our key goals. Besides the research, we could acquire the *Handelsblatt Research Institute*[1], the research lab of an award-winning online editorial counting to the top 10 news providers in Germany, as a partner in January 2020 to use our software in their articles. By this we try to gain in the nearer future deeper knowledge and a better understanding how real users in a non-laboratory environment interact with online-articles.

1.2.3 Arguments as a Resource

During the process of developing and researching, we changed the way we think about arguments. Using arguments in discussions is essential. But one of the key questions, which needs to be solved, is how to improve the redundancy in discussions. And transitioning arguments from ephemeral strings to structured *data* could lead to a significant improvement.

Arguments should be considered as *valuable data*, which can be reused in conversations. We are focusing on this in Chapter 5. We present first research on reusing (previously discussed) arguments in Schneider and Meter 2017. This paper discovers new ways to handle arguments in a distributed environment, like the Internet, and how to merge changes or discussions to these arguments. Using these ideas, we are proposing a versioning system for arguments, inspired by the git-protocol[2].

Afterwards, we present a definition and a reference implementation for *Extensible Discussion Entity Network* (EDEN) (Meter et al. 2018c). This network is used to exchange arguments and their relations in a federated peer-to-peer network.

Summarizing our research in the field of DBOA, we published in Schneider and Meter 2019 a paper describing all our efforts and also published our software under an Open-Source license on GitHub[3].

1.3 Outline of this Thesis

Preliminaries showcasing the current tools, which are used for online discussions, are described in Chapter 2. Also, some terms from argumentation theory can be found in this chapter. Chapter 3 contains a short introduction into DBOA and presents the tool D-BAS as well as first field experiments with untrained users to evaluate our novel discussion approach. Afterwards, Chapter 4 explains our thoughts about embeddings of DBOA-tools into web context for tighter interactions with online articles and to enable structured discussions wherever the Web comes into play. Our tool discuss and the final study evaluating our complete software stack is also described in this chapter. Chapter 5 follows with the papers about considering arguments as a resource, which should be further distributed to make them accessible in other discussions.

[1] `https://research.handelsblatt.com/en`
[2] `https://git-scm.com`
[3] `https://github.com/hhucn`

At last, the dissertation closes with Chapter 6 containing the conclusion and more outlooks on further research.

Chapter 2

Preliminaries

To understand problems with current online discussions, we first have to take a look on the current state of software tools or rather in what categories they can be placed. This chapter concentrates on the basic building blocks of current discussion tools and then describes our terms concerning argumentation theory or how we define them.

2.1 Unstructured Data

There are few categories that current tools can be divided into, regarding the way they encourage online discussions. The following paragraphs give a general description of these categories.

Forums A typical forum contains several threads, where each thread has a question, issue or problem in its focus. A user describes the problem and starts the discussion. Following this, it is possible to post comments in this thread which are then usually presented in a list. It is commonly feasible to reply to a comment with a post, see Figure 2.1.

First prominent examples of this approach for discussions are electronic mails or the *Usenet*. Modern implementations, like Reddit, phpBB or bulletin-board-like software, are nowadays commonly seen and heavily used when it comes to online discussions or e-participation (Sæbø et al. 2008).

These implementations are easy to use and widely spread, but giving the users the possibility to write in a text area whatever and however they want leads to an unstructured nature of the posts. Ideally, the posts are semantically referring to the current thread's topic, but there are no restrictions in what the users have to write. Therefore, automatically extracting the structure of a discussion (which post attacks or supports another) is currently heavily being researched (Cabrio and Villata 2018; Lawrence and Reed 2020). If the participants really want to have a constructive discussion, as it can be seen in Reddit's `/r/ChangeMyView`[1], then forums are quite useful. Even changing someone's opinion, which can be challenging and tough using

[1] https://www.reddit.com/r/ChangeMyView/

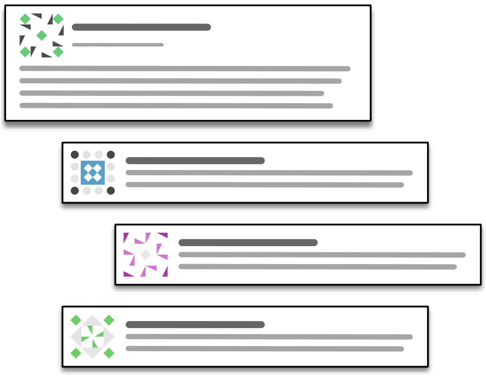


Figure 2.1: Schematic view of a forum-based discussion. The indentation represents the reply structure.

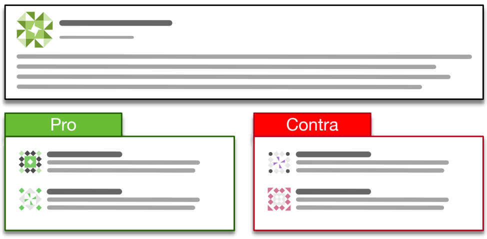


Figure 2.2: List with one proposal and some pro/contra arguments.

only arguments, can then be achieved (Tan et al. 2016). This underlines the usefulness of these kinds of tools.

Pros and Cons Lists Threaded discussions with the distinction between pro- and contra-opinions are the way pros and cons lists encourage an exchange, see Figure 2.2. This approach is often being used when it comes to decision-making processes in the political context, for example in Iceland (Lackaff 2016) or Germany (Liebeck et al. 2016; Steinbach et al. 2019).

Having a threaded forum-like discussion with the possibility to sort new (sub-)threads into the pro- and contra-scheme, gives at least a bit more information about the structure of the discussion. But within a sub-thread, there are still listed comments in free text form, so we have the same disadvantages we have seen in forums.

Similar Approaches More approaches (e.g. comment sections) can often be reduced to these categories. But they all have in common that the results are masses of text entries, lacking relations to other entries most of the time, which makes it difficult to extract the argumentation structure or the coherence between the posts.

Having to read all comments or posts does not scale well, because for each participant joining the discussion, all other users have to read the arguments from the new user. This is illustrated in Figure 2.3.

Argumentation Systems An *argumentation system* in general is a system containing a set of rules guiding the participants through a dialogue. In this work, we focus on argumentation systems, which are commonly software applications. These applications either guide the par-

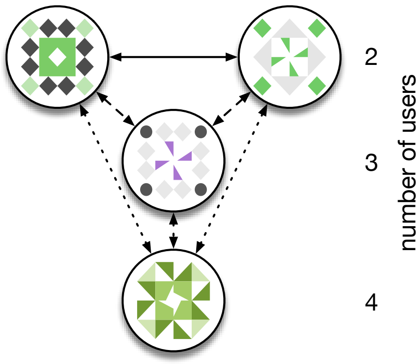


Figure 2.3: Users in a discussion. The connections illustrate how many new argument sources must be inspected when a new user joins to keep an overview in a discussion. Figure based on Ebbinghaus 2019, p. 5.

ticipants through a stored discussion as some kind of "playback" or interactively collect the arguments following the set of rules.

Argument Maps A different scientific approach are *Argument Maps*. These maps are often manually produced and represent a two-dimensional argument structure (Macagno et al. 2007; Walton 1996, chapter 6). Argument mapping is commonly being used in software tools using *Computer-Supported Argument Visualization* (CSAV) (Shum 2003), which aim to support the understanding of a discussion by separately drawing premises, conclusions and their relations. They are even used to enhance critical thinking (Twardy 2004) or policy-making (Renton and Macintosh 2007).

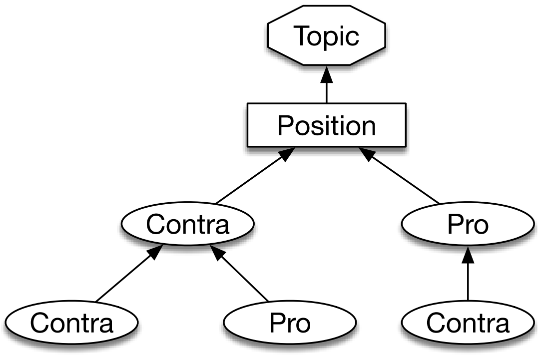


Figure 2.4: Sample argument map. Depicted is a discussion topic, a sample position and some pro/con statements.

Using these maps supports understanding the discussions, but oftentimes the maps have to be created manually. Our goal is to automatically produce argument maps, which are derived from the arguments being collected by our software systems. See Lawrence and Reed 2020, pp. 13–15 for further examples on argument maps and more detailed explanations.

2.2 Argumentation Theory

Researchers from different research fields are working on argumentation theory which is why there are several interpretations and definitions being used. Since they might differ in small nuances, we are giving a short overview of our definitions. If there are more definitions needed, they will be provided in the separate papers included in this thesis. Most terms are consistent with the definitions described in Lawrence and Reed 2020; Pollock 1987; Prakken and Vreeswijk 2001.

Terms

Statements

The smallest building blocks in our computer-scientific view on argumentation theory are statements. The negation of a statement itself is also a new statement. They are (usually) user-generated and used in discussions. Statements receive their value by the contexts in which they are being used.

Positions

A statement which is used as a discussion opener, e.g. proposals for action, is called *position*.

Premises and Premise Groups

If statements are used to support another statement, then they are called *premises*. One or more premises form a *premise group* if used together.

Conclusions

A statement, which is supported or attacked by a premise group, is called *conclusion*.

Relations and Inference Rules

Relations, or *inference rules*, between conclusions and premises can either be supportive or attacking.

Arguments

Now that we have our building blocks, we can define an *argument*. Arguments consist of a conclusion, a premise group and a relation.

Attacks and Supports

Users can attack or support parts of an argument. There are three possible attack-types and three support-types, all of them have one part of the argument as a target: premise group, conclusion or relation. Some attacks have special terms. Attacking the premise group is an *undermine*, attacking the conclusion is called *rebuttal* and attacking the inference rule of an argument is called *undercut*.

Topic

A set of arguments can be collected to a *topic*. This is usually done when people are discussing, but this is not a technical requirement.

2.3 Related Work

This section deals with related work from academic projects and some examples of non-academic solutions for achieving structured discussions on the Internet.

Argument Maps There are several systems introducing structured discussions on the Internet, that have been developed in the past years. First systems gather free-text and enable manual selection of text passages to mark them as some argumentation entity. *Araucaria* (Reed and Rowe 2004) started in 2004 with an interactive application, where these selections can be made. *Carneades* (Gordon and Walton 2006) and *ArguNet* (Schneider et al. 2007) both build upon the idea of Araucaria. More recent versions of online argument-mapping tools are *OVA+* (ARG-tech 2020; Janier et al. 2014) and *MindMup* (Sauf Pompiers Ltd. 2020), which are highly interactive web applications supporting live annotation of discussions. All mentioned applications can be considered expert tools allowing trained users to directly collect the arguments from a debate or some texts and create a live argumentation map.

Argument-mapping systems usually target professional users and give them the opportunity to understand a discussion in more depth. Since we are targeting unskilled participants, these systems are only related due to their creation of an argument map. Nevertheless, we differ in the way argument maps are built. We want to enable *automatic* and *live-building* of argument maps so that users may use the generated overview, but do not have to create them on their own.

Pros and Cons Lists, Comments and Forums Pros and cons lists are an easy approach to enable the presorting of arguments. Usually, conversations take place via a comment section, where users need to define whether to support or attack a proposal in the first place. After this, the arguments can commonly be attacked or supported in a threaded style we already saw in forums. A popular example in the academical context is *ConsiderIt* (Kriplean et al. 2012, 2011), allowing the users to position themselves by using sliders indicating how much they support an argument. The *Deliberatorium* (Klein 2011), another popular example, uses techniques from forums, lets participants comment on arguments and includes a manual rating of arguments. There are many more systems freely available on the Internet, for example *Kialo* (Kialo 2020) or *GroupMap* (GroupMap Technology Pty Ltd. 2020). Productive debates in a forum-based system can be found in *Reddit's ChangeMyView* subreddit (Reddit Inc. 2020), where moderators are used to keep a line in the discussion. Summarizing, it can be said that countless approaches for forums, pros and cons lists and comment-based systems can be found on the Internet as of now.

These systems are united in so far that free-text is their main source of input, providing the option to sort their arguments either based on their depth in the discussion or by proposing an initial attitude. Common problems, as described in Section 2.1, usually occur with these approaches. Their simplicity enables even untrained users to participate in online discussions, explaining their high prevalence. Despite this fact, lack of argumentational structure is still a problem when it comes to the analysis of the content. Therefore, we keep these systems in mind when designing and researching simple argumentation systems, nevertheless we still need to enhance them in such a way that they readily provide or enable one to easily extract the discussion's structure for further processing.

Issues, Positions and Argument Structure Several argumentation systems building on (different) computational models of argumentation appeared in the last years. The basic theoretical ideas are presented in the *Issue-Based Information System* (IBIS) (Kunz and Rittel 1970). First implementations rely on free-text and provide options to explicitly declare the own argument as supportive or attacking. *MAgtALO* (Wells and Reed 2007, 2008) presents a chat-system to the users, where each comment can either support or attack the proposal. The *Structured Consultation Tool* (Wyner et al. 2011) constructs and presents surveys to gather feedback from the users, which is then structured. Another prominent system is *Arvina* (Lawrence et al. 2012a), which can be used to ask experts questions from a pre-conducted discussion or to have new conversations. Arvina then builds the arguments on the fly. More approaches enabling structured discussions are *DEMOS* (DEMOS 2020; Lührs et al. 2001) for e-participation or *HERMES* (Karacapilidis and Papadias 2001) for decision-making processes.

These systems all have in common that they influence the way arguments are inserted. The user is explicitly asked for some context in order to find the correct place in the argument map. In our research, we share the same vision and ask the participants concretely which part of an argument they want to attack and how their current attitude can be described best. Asking for necessary context reduces problems when it comes to interpretations during the processing phase of the arguments, which is why we want to create our own systems following this idea of processing data.

Embeddable Discussion Structure Argumentation systems have in common that they either provide a dedicated software or a standalone website accommodating the functionality to discuss online. Leaving the scope of a single website to bring discussion functionality in arbitrary contexts is implemented by *Disqus* (Disqus, Inc. 2020). The software is easily embeddable in websites and provides a plug-in comment-section. However, conversations within Disqus do not exhibit additional structure in comparison to forum-based systems, but are easy to use for untrained participants. Interactions with online articles are possible with tools like *rbutr* (rbutr 2020) or *ArguBlogging* (Bex et al. 2014b). With the latter it is possible to collect content from online articles and comment it on a blog resulting in a comment-section-style conversation. On *medium.com* (Medium Corporation 2020), users can publish their own articles, which other users can discuss by selecting text passages. With these selections, participants can react to specific parts of an online article and newer users can then see these passages at the bottom of the article.

Many academic prototypes in this field are no longer being maintained. Commercial products, like Disqus or medium.com, are successful tools, which are used by many people. Especially the lightweight integration of Disqus and the deep interaction possibilities of medium.com are promising techniques that can be useful in our research field. We will take a look at these techniques when it comes to the embedding of dialog-based discussions into arbitrary web-contexts.

Corpora Databases After collecting the arguments the users exchange, we have to think about storing and distributing the obtained data. A first huge corpus of arguments is stored in *AraucariaDB* (Reed 2006), which presents discussion data in an accessible database. The successor of this database is *AifDB* (Lawrence et al. 2012b) replacing and enhancing the basic idea of AraucariaDB. Both databases are parts of the *Argument Web* (Bex et al. 2013; Rahwan et al. 2007), depicting the idea of infrastructure solely for argumentation. A more recent argument database is provided by *args.me* (Wachsmuth et al. 2017) containing corpora of rated arguments. Queries to this database provide arguments sorted by relevance and put into a pros and cons list. More online databases, oftentimes combined with the functionality to discuss in a comment-style or via a pros and cons list, are providing their corpora of arguments. Popular examples are *Debatewise* (DebateWise 2020), *IDebate* (idebate 2020), *Debatepedia* (Debatepedia 2020) or *debate.org* (Debate.org 2020).

All databases are united by the idea of an argument-interchanging network. We take up this idea, because databases storing the arguments from several discussions are the base to reusing previously discussed arguments. This may reduce redundancy in current debates, because arguments do not have to be discussed over and over again. Therefore, we extend the idea of a specific database and add a distributed data structure. In doing so, arguments are not stored on a central server, but can be stored anywhere and exchanged via the Internet.

Chapter 3

Dialog-Based Online Argumentation

Dialog-Based Online Argumentation (DBOA) is the key building block for our research. At first, we had to understand how people are discussing in the real world. Secondly, we start to propose new ways to discuss on the Internet.

This chapter contains a short introduction to our goals, our research papers about DBOA, D-BAS, a field study and an experiment in the context of decision-making.

A key goal of our research is to transform the generated discussion data into structured data, to be analyzed and automatically used by algorithms. We achieve this by defining a *feedback-cycle* (see Figure 3.1), which the participants pass through, while we collect their arguments in each iteration.

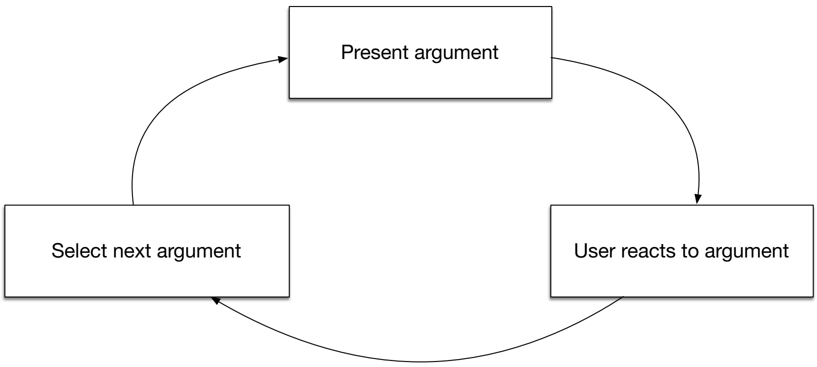


Figure 3.1: Visualization of a feedback-cycle in the dialog-based approach. Users start at “Present argument” Graphic based on Figure 1 in Krauthoff 2018.

The arguments are selected and presented to the participants by our DGEP. In the following, we will always use D-BAS as our argumentation backend.

Now, let us focus on the complete procedure of discussing in the original dialog-based fashion. First, the users select a topic and are presented with the positions provided by other users, or, if there are no (interesting) positions, the users can add their own ones. After this, users have to choose whether they agree, disagree or have not yet formed an opinion regarding this position. After this step, users see arguments from other users and may now choose to follow the other users' opinions or to provide their own input to the discussion.

Afterwards, users are confronted with opposing arguments from other participants and can react to these arguments. They have the options to attack the conclusion, premise(s) or relations of the opposing argument, or they are convinced by the opposing argument and accept it (see Section 2.2 for detailed information about the reaction options). The feedback-cycle starts again by presenting a different argument and asking for the users' opinions.

We are using *natural* and *easy* language to guide the users through the discussion. This means that there is no need to have a deep understanding of argumentation theory, because all steps are explained.

As a result, we generate *structured data*, which can be used for further processing. This data can be represented as an argumentation map (see Section 2.1), and is being rendered by D-BAS' graph visualization engine, see Figure 3.2. We can see all the parts of the discussion, and infer the structure of the discussion by guiding the user and asking the right questions.

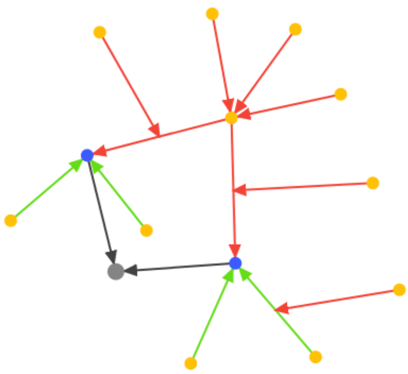


Figure 3.2: Such graphs are created from discussions through D-BAS, discuss and Jebediah. Depicted is an instance from a real-world discussion. Colors: ● issue, ● positions, ● statements, ▬ supports, ▬ attacks (Krauthoff et al. 2018, 2017)

After a discussion is finished, we can walk through the argument map to check the relevant positions, generate summaries from the sub-discussions and analyze the graph structure. We are currently working on more automatic procedures to get statistics and information about the generated graph and the containing arguments.

In classical approaches, people have to manually find the arguments in the free-text the participants wrote to classify and structure them. This is not an easy task as we experienced in our studies (see Section 4.3), because the arguments in an unstructured text do not follow any recognizable pattern — different users have different ways to discuss. This is also a problem when it comes to mining the arguments from discussions with algorithms (Budzynska et al. 2014; Lippi and Torroni 2016). Our approach leaves these problems behind, because the guided dialog allows structured recording of arguments directly when the users are writing their statements. No interpretation and no further post-processing of the collected data is necessary, which is our unique selling point.

In the following sections, the papers about this research are being described. It starts with the general approach of DBOA, followed by some studies. The next chapters build upon this idea and provide further information and research findings about using and integrating dialog-based discussions in the real-world.

3.1 D-BAS — A Dialog-Based Online Argumentation System

This section gives an overview about the contributions and impact of our paper Krauthoff et al. 2018:

Tobias Krauthoff, Christian Meter, Michael Baurmann, Gregor Betz and Martin Mauve.

"D-BAS — A Dialog-Based Online Argumentation System"

In: *Proceedings of the 7th International Conference on Computational Models of Argument*, Volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 325–336, IOS Press.

Acceptance Rate: ∼50%

A first version of DBOA and D-BAS was published as a short paper in Krauthoff et al. 2016. It describes the first ideas of the dialog-based discussion system D-BAS and gives a brief overview about the argumentation system. A more mature version with several improvements was than published with this paper describing the dialog-based online argumentation approach in full.

Summary

In this paper we describe the current state of online deliberation processes at a local level, where the goal was to involve the citizens' opinions in political decision processes. The problems described in Section 1.1 are linked to the real world and explained. Other systems, which try to solve these problems, are named. But these systems can often be reduced to the already known variants, e.g. forums and comment sections, which are described in Section 2.1. Afterwards, we give a brief overview of our definitions of terms that are used in argumentation theory.

The paper's Chapters 4 and 5 describe our solutions about text-generation and user interaction. As a general problem, we had to solve how to present the arguments of the discussion to the users. It is simply not possible to show all arguments to them, because then we will not achieve better results than forums or pros and cons lists. So we developed the dialog-based approach, where users are presented with an argument, gathering the reaction of the user and then presenting the next (attacking) argument. Following this cycle seems easy enough to guide users through the discussion and presenting them relevant arguments.

Reactions to arguments were reduced by us to those, which are being used in real discussions. For example, we dropped support options regarding the inference rule of an argument, because in a real discussion, users are typically not supporting another argument by adding a new statement, which underlines how excellent the premises are supporting the conclusion. Our reductions of the feedback options to the more relevant ones is being shortly described in this paper and in more detail in a field study with real users (Krauthoff et al. 2017).

Navigating the WoR and choosing a fitting next argument is a problem which needs special attention. We tested two approaches with several users: (1) choosing an attacking argument, which is opposing to the user's opinion or (2) mix supportive and attacking arguments as the next candidate. (2) was not a suitable fit, because adding a supportive argument to a previously stated argument from a different user does not encourage the users to discuss lively. Therefore, we chose (1) in the current implementation of D-BAS. The detailed results are also described in (Krauthoff et al. 2017).

At last, we describe a *Decentralized Moderation System* (DMS), which allows moderation without dedicated moderators. In a DMS, the participants can decide in a democratic process if some statements need special attention, should be deleted, or similar. By this, our argumentation system becomes self-maintained with nothing more than the participating users.

The initial idea of DBOA was introduced by Martin Mauve and Michael Baurmann. Gregor Betz provided the first ideas about argumentation structure. But Krauthoff et al. 2016 and Krauthoff et al. 2018 provided the first whole view on this novel approach.

Personal Contribution

Christian Meter, one of the contributing authors, jointly maintained the code base with Tobias Schröder (né Krauthoff). Christian provided ideas like the color-coding substitution of long sentences, optimizing interactions with participants, text-generator optimizations or the DMS and described them in the paper. He spent extensive work on the modularization of D-BAS' argumentation logic (the DGEP) to open the core logic for new applications via an API (see Chapter 4) and added these sections to this paper. The base project as of 2015 has been developed by Tobias. Also, most of the features, except the API, were initially implemented by Tobias, but afterwards jointly maintained. Writing descriptions about the technical details (Section 8) and the field experiment (Section 9) as well as reviewing the rest of the paper were done by Christian.

Importance and Impact on this Thesis

Being the first paper describing the idea behind DBOA and the argumentation system D-BAS in full, it is one of the most important papers for this thesis. The core argumentation logic is being used in most of the following papers and still represents the basic building block of our research on the way to create better online discussions. Also, first results of experiments are summarized in this paper, which underlines the practical usability of this approach.

D-BAS - A Dialog-Based Online Argumentation System

Tobias KRAUTHOFF [a,1], Christian METER [a], Michael BAURMANN [b], Gregor BETZ [c] and Martin MAUVE [a]

[a] *Department of Computer Science, University of Düsseldorf, Germany*

[b] *Department of Sociology, University of Düsseldorf, Germany*

[c] *Department of Philosophy, Karlsruhe Institute of Technology, Germany*

Abstract. In this paper, we present D-BAS, a dialog-based online argumentation system, tailored to support e-participation processes. The main idea of D-BAS is to let users exchange proposals and arguments with each other in the form of a time-shifted dialog where arguments are presented and acted upon one-at-a-time. We highlight the key research challenges that needed to be addressed in order to realize such a system, provide solutions for those challenges, report on a full scale implementation of D-BAS and summarize the findings from a real world e-participation process, where D-BAS provided the infrastructure for online argumentation.

Keywords. online argumentation, dialog-based argumentation, online participation

1. Introduction

E-participation, such as urban planning or participatory budgeting, is an important application area for online argumentation. In these processes citizens that will be affected by future decisions are invited to participate in the decision making process by proposing actions and discussing them with their peers. The results of the discussion, i.e., the proposals and the arguments, are then incorporated in the decision making process.

E-participation is a challenging application area for online argumentation, since the participants might be experts in the problem domain, but they are not experts in argumentation. Additionally, they often have a significant stake in the topic that is being discussed. Therefore they typically want to convey their point of view rather than engage in evidence-based deliberation. Furthermore, the number of individual contributions can be very large in these processes, while at the same time, the available resources are often rather limited. As a consequence it is frequently not an option to have experts in argumentation involved in the process in order to take the input of the participants and then structure it in an appropriate way.

At the same time, however, the result of the online argumentation is not the final outcome in an e-participation process. Instead, it is taken as an input by those that finally make a decision, such as elected representatives. Thus there is a layer above the online

[1]Corresponding Author: Tobias Krauthoff, Department of Computer Science, Universitätsstraße 1, 40225 Düsseldorf, Germany; E-mail: krauthoff@cs.uni-duesseldorf.de

argumentation that is able to interpret and weight the individual proposals and arguments. It is therefore not necessary that online argumentation in e-participation processes arrives at a certain conclusion such as a consensus or any other form of decision. Instead, a well structured set of proposals and (interrelated) arguments is a perfectly acceptable result.

So far, online argumentation in e-participation processes mainly relies on forum-based systems. This leads to well known problems such as limited scalability and a lack of structure [1]. As a consequence there have been several attempts to provide better support for online argumentation. However, so far, none of them has had really significant practical impact. One important reason for this may be that forum-based systems offer something that other systems do not: they allow for a highly complex exchange of arguments and counter-arguments with an intuitive statement-reply-scheme. Other approaches to online argumentation either do not capture the full complexity of argumentation (e.g., pro/con lists) or they require that the user is trained in operating a rather complex technical tool (e.g., the cooperative creation of an argument map).

In this paper we present D-BAS, a dialog-based online argumentation system, that does not require any prior knowledge or training from the user and avoids the shortcomings of forum-based systems while still allowing for complex argumentation. The main application scenario we have in mind for this approach is e-participation, while we do believe, that it might be applicable to other areas as well.

The key idea of our approach is to guide participants through the arguments provided by other users so that they perform a time-shifted dialog with those that have participated before them. The system is driven by a formal data structure capturing the full complexity of argumentation. User interactions, however, have the structure of a regular dialog as it is performed in everyday life. It is the task of system – and not of the participants – to translate between those two views. We call this approach *dialog-based online argumentation*. The output of dialog-based online argumentation is a set of proposals and interrelated arguments, both provided by the participants. While it might be possible to extend dialog-based online argumentation to include group decisions, this is not part of the work described here.

This paper is structured as follows. In Section 2 we give a brief overview of related work in the area of online argumentation. Section 3 explains the model of online argumentation used by D-BAS. The general idea of dialog-based online argumentation is outlined in Section 4. The solutions to key challenges are presented in Sections 5, 6 and 7. In Section 8 we describe the implementation of D-BAS as a fully functional dialog-based online argumentation system. We show that the idea of dialog-based online argumentation is viable by summarizing the findings from a deployment of D-BAS in a real world setting in Section 9. The paper is then concluded by a summary in Section 10.

2. Related Work

The general idea, key challenges of dialog-based online argumentation and details about our argumentation framework were already given in [2]. We have analyzed the details of a real world deployment and evaluation of D-BAS in a paper presented at AI[3] 2017 [3]. The current submission, on the other hand, only briefly summarizes the key findings of the real world deployment in order to show that the idea of dialog-based online argumentation is actually viable and focus developing a fully fledged dialog-based online argumentation system called D-BAS, targeting e-participation processes.

Current approaches for online argumentation in e-participation processes can be roughly separated into three main groups: forum-based approaches, pro and contra lists and argumentation maps.

The first group, also called asynchronous threaded discussions, allows participants to exchange arguments by means of a sequence of text contributions. Those approaches have encountered much criticism in the past, because they are believed to lead to a high degree of redundancy [4] and polarization [5, 6] while scaling poorly with the number of involved participants and supporting non-collaborativness [7]. However, in practice they are, by far, the most commonly used approach to support online argumentation.

It has been suggested, e.g. ConsiderIt [8], to use online pro and contra lists to aid collective decision making processes. These lists work very well for evaluating a given proposal. However, they are not suitable to deal with more general positions and alternatives since they do not support the exchange of arguments and counter arguments.

Online systems for argument mapping enable participants to structure their arguments and the relation between them in an argument map. Examples are Carneades [9] and Deliberatorium [10]. While those systems do avoid the shortcomings of forum-based approaches, they require the users to become familiar with their notations and the semantics of formal argumentation. Therefore, in practice, they are used by experts or students who want to learn about the logic of argumentation rather than by average users that want to take part in an e-participation process.

The idea of engaging in a formalized dialog to exchange arguments is used by so-called dialog games, which follow a set of rules to react to the statements of each other, see [11]. In contrast to our work, dialog games focus on the real-time interaction between users in order to learn something about a subject at hand. They do not seek to provide better instruments for online argumentation.

In addition to general work on online argumentation there are three individual systems that are related to our work. The first one is the *Structured Consultation Tool* (SCT) [12]. Its primary goal is to allow a government agency to elaborate and present a justification for a given action. While the SCT explicitly seeks feedback on the arguments provided by the government agency, it does so in a questionnaire kind of way. This is valid for gathering feedback on government proposals, but it is unsuitable for an online argumentation, where the dynamic exchange of arguments is the main focus.

The Carneades Opinion Formation and Polling Tool [13] is part of the Carneades argumentation mapping system. It allows participants to provide structured, questionnaire-style feedback on a given argumentation consisting of multiple arguments and positions put forward by – potentially – many agents. This tool can be regarded as a generalization of the SCT. As with the SCT the questionnaire-style feedback is well suited for an evaluation of government activities by citizens but it does not fit the idea of an online argumentation amongst peers.

The third system that is related to our work is Arvina [14] and its predecessor MAgtALO [15]. Both systems allow a user to conduct a dialog between robots and humans. As a basis, they use an existing discussion where the positions and arguments of some real-world persons are marked. In contrast to D-BAS Arvina and MAgtALO are driven by the questions of the users. Thus there is no need for the users to react to replies from the system by providing their own arguments.

3. The System Perspective

In the following we assume that every online argumentation is identified by a *topic* that describes what the argumentation is about. *Statements* are the most basic primitives used in an online argumentation. Individual participants might consider a given statement to be true or false. The negation of a statement is itself a statement. A *position* is a prescriptive statement, i.e., a statement which recommends or demands that a certain action can be taken.

We distinguish between first-order and second-order *argumentation*. First of all, there is argumentation for or against statements. Here, some statement (the premise) is said to be a reason for another statement (the conclusion). This leads to a first-order argument, consisting in a premise-statement, a conclusion-statement and a reason relation between both. With this structure it is straightforward to represent undermines and rebuttals. A first-order argument *A* attacks another first-order argument *B* iff *A*'s conclusion is the negation of a premise of *B*; and *A* is a rebuttal of *B* iff *A* and *B* have contradictory conclusions.

Still, we must not presuppose that untrained participants in dialog-based online argumentation advance only deductive reasons and valid arguments. The reason-relations claimed by users might be more or less cogent – and more or less evident for other users, which may trigger a discussion about the strengths of reason-relations. That is what we call second-order argumentation. Accordingly, we allow reasoning not only about statements but also about whether one statement really supports (attacks) another statement. A second-order argument consists of a statement (the premise) that is cited as a reason for why another reason-relation does not hold. Second-order arguments allow us to express undercutting attacks, namely as arguments against reason-relations pertained in other arguments.

As a consequence we use the following definition: an argument consists of one or more statements, which form the premise(s); one statement or the second-order claim that a certain reason-relation does or does not hold, which forms the conclusion; and the reason-relation between premise and conclusion. Together, the arguments of a debate form a (partially connected) *web of reasons* (WoR).

We would like to stress that our data structure and the distinction between first-order and second-order argumentation in a user dialog does not commit us to a specific argumentation-theoretic framework. On the contrary, the dialogs we model can be interpreted in quite different ways:

- *Deductive argumentation:* The arguments we model can be understood as enthymemes, i.e. incomplete arguments, that can in principle be reconstructed as deductively valid arguments if all implicit assumptions are made explicit. On this view, second-order argumentation would actually be argumentation about the plausibility of those implicit assumptions.
- *Probabilistic reasons:* The reason-relations can be explicated in probabilistic terms. On this view, a second-order argumentation undermines or establishes the probabilistic reason-relation maintained in another argument.
- *Defeasible reasons:* The arguments we describe can be understood as defeasible reasons. On this view, a second-order argumentation defeats another argument (or attacks a defeater).

We conceive this theoretical openness of our argumentative dialog model as a major strength. Note that the data we generate can also be used to check how well the alternative paradigms of rational argumentation can cope with the discussions we protocol.

4. The User Perspective

The foundation of dialog-based online argumentation is a novel way to navigate an existing set of arguments pertaining to a given subject. Instead of presenting many arguments at once – in maps or lists of arguments – the user is shown only a single argument at a time. It is then possible for the user to respond to that statement, either by selecting a statement provided by another user or by entering a new statement. Based on this response and, possibly, the data gathered from the responses of other participants, the system selects the next argument that is shown to the user. In this way the user and the system perform a dialog where the system selects arguments that are likely to be of interest to the user and the user provides feedback on those arguments.

Both, the user and the system, profit from the dialog. The user is efficiently guided towards those arguments that are particularly relevant for her. This also reduces redundancy, polarization and the occurrence of logical fallacies. The system, on the other hand, will increase its knowledge base with every response from a participant. This can then be used to improve the selection of arguments for the next user and to provide a summary of the online argumentation at hand.

5. The First Challenge: Feedback

The most basic building block of dialog-based online argumentation is gathering feedback from a user regarding a given argument. This is done by asking a question derived from the statements pertaining to the argument in the WoR. For example, if we have a premise-conclusion structure, the question generated by the system would be "What do you think about the following argument: ...?". The system then offers a set of answers from which the user can choose. This set has to be constructed in a way that enables an untrained user to provide precise feedback on the argument. A simple choice between: "I agree with this argument" and "I do not agree with this argument" could undoubtedly be made by an untrained participant. However, both statements are not precise and have little significance. For example "I do not agree with this argument" might refer to several distinct scenarios: the user might disagree with the premise, the user might think that the conclusion is not supported by the premise or the user might consider this to be a valid argument, but at the same time she might consider, that it is weaker than other arguments supporting the negation of its conclusion.

In order to get precise and meaningful feedback from the user, the system has to differentiate between the scenarios by means of a set of answers that the user can choose from. Experiments with a prototype system that allowed users to react to arguments of a pre-constructed online-argumentation led us to one key observation: giving feedback on an argument is a two step process. The first step requires just a single click from the user to determine her initial reaction to the argument, e.g. the user rejects the premise of the argument. As a second step the user can then provide a justification for her choice either by selecting an existing statement or by providing a new one. Separating the two steps facilitates very fast feedback and a clean design of the user interface.

Next, we constructed the set of reactions that would be offered to the user in order to respond to a given argument. As a basis we used attacks as they are defined in argumentation theory and added two options that are frequently used in informal argumentation. This led to the following set: (1) Reject the premise. (2) Accept the premise and, as a consequence, the conclusion. (3) Accept the premise but disagree that this leads to accepting the conclusion. (4) Accept the premise but state that there is a stronger argument that leads to rejecting the conclusion. (5) Do not care about the argument.

Once the user has selected an answer and provided a reason, the system uses this to update the internal information of the WoR and to select the next argument that is presented to the user.

5.1. Optimizing the Representation of Questions and Answers

In an early implementation of our system we simply repeated the full *premise* and *conclusion* for each individual option that the user could choose. First tests showed that this leads to very long feedback options where some text parts were repeated several times. Participants in those tests indicated that this was not acceptable since they lost their focus when reading all the feedback options. As a solution to this problem, we use terms like “my point of view”, “their statement” or “their point of view” instead of repeating the actual premise and conclusion of the argument. In order to make sure that the participants can easily determine what those terms refer to, both the terms and the premise or conclusion they refer to are colored in the same way. An example is shown in Figure 1

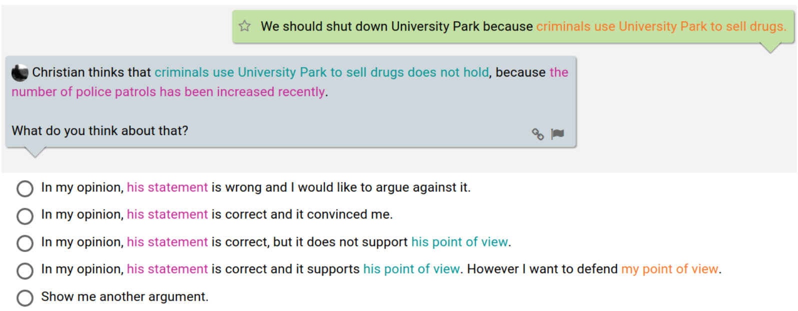


Figure 1. Challenging the user’s argument and getting feedback from the user.

5.2. User Evaluation

We conducted two experiments with 18 participants each. The goal of the experiments was to verify whether we have included all relevant feedback options. For the first experiment we used the initial feedback options without substituting the premises and conclusions. The second experiment then employed those substitutions including coloring and highlighting. In both experiments the number of male and female participants was about the same and the age of the participants covered a wide range.

Both experiments were designed as follows: the participant was shown two arguments. A first argument represented the opinion of the participant and a second argument was an attack on the first argument. The participant was then asked: “What is your opinion regarding the second argument?”. The participant would then write down her answer in a simple text field. Afterwards, the participant was shown the feedback options described above and asked the question: “Which option would you choose?”. Comparing the text entered with the chosen option allowed us to determine whether the user is able to indicate the desired reply by using the feedback options.

The first, and possibly most important, result of the experiment was that every single reply given in the text field could be matched to one of the feedback options we described above. This indicates that our feedback options are complete. We then investigated whether the participants provided consistent answers and thereby showing that the feedback options were presented in a way that the user would understand.

In general, the user answers could be mapped to the appropriate feedback option with a chance of 72.2% for the first experiment. Especially in the second experiment there were less differences between the free text and the chosen feedback option, because the mapping increased to 83.3%. We believe that the improvement is caused by a reduction in the complexity of the feedback options when using color-coded replies.

6. The Second Challenge: Navigating the Web of Reasons

The second major challenge for dialog-based online argumentation is how the system should select the arguments that are presented to the participant. We address this challange through three mechanisms: (1) bootstrapping the dialog by identifying the first argument that should be presented to a new user; (2) selecting the next argument based on the prior actions of the user and (3) an optional lateral entry into an ongoing argumentation.

6.1. Bootstrapping

The first thing that the system usually needs to do when a new user wants to participate is to choose an initial argument to present to the user. This is challenging since the system has no information on the user, yet. We solve this problem by asking the participant which initial position she is interested in. This position is selected as the starting point in the WoR. The user is then invited to indicate her attitude towards this position: she can support or attack the position.

After the supporting or the attacking option is chosen, the user is asked to select or provide a statement explaining her choice. This statement is used as the premise, whereby the position (or its negation) is the conclusion. This completes the first argument and ends bootstrapping.

6.2. Selecting the Next Argument

The selection of the next argument that is presented to the participant can be based on different selection strategies. We have chosen a simple antagonistic strategy that mimics typical human behaviour in an argumentation: we look at the participation history of a user to identify the most recent argument that she selected. Then we search the WoR for an argument of prior users which attacks (undermine, rebut or undercut) that argument. This argument is shown to the user who then has the opportunity to react to it. This process continues until the WoR contains no counter argument to the argument of the current user. The overall intention is to simulate a real discussion where participants challenge the arguments of other participants.

6.3. The Quick Lateral Entry

When a user has already particpated in a discussion she should be able to navigate directly to an argument instead of starting from scratch. To this end, we provide two additional means of navigation: First, the user can search for any statement via a query mask. As outlined above she is then invited to indicate her attitude towards this statement, completing the alternative bootstrapping process. Second, the underlying data structure can be viewed as interactive argumentation map. After the selection of a statement, the system displays all arguments connected to this statement. The user is then invited to select any of these arguments.

7. The Third Challenge: Accepting and Maintaining Arguments

The key to incorporating new arguments in dialog-based online argumentation is to nudge the users to provide arguments in an appropriate way. Currently, we use three mechanisms for this purpose.

First, users can enter their own statements only within the dialog, so that given statements automatically are connected to the WoR in an appropriate fashion. Second, we apply sentence openers to frame the statements of the users. In this way the user is guided towards making structured and well-formed statements. Third, we automatically match the text entered by a user with existing statements in the WoR by means of Elasticsearch[2]. This reduces redundancy if the user chooses to use an existing statement instead of providing a new statement. An example of statement proposals during the users input as well as the sentence opener is given in Figure 2.

While those mechanisms improve the quality of the arguments provided by the users, they can not prevent that a given user input is incorrect. To address this problem we use a decentralized moderation system, so that the every participant is able to review statements and propose improvements by means of review queues. If one user flags a statements due a specific reason, other users can go through those queues and vote on the action to be taken. Once a sufficiently clear-cut collective opinion has been reached, the appropriate action is taken, e.g. the statement might be replaced or deleted or the flagging might be discarded. Based on our experience with dialog-based online argumentation we suggest the following review queues:

- *Delete:* Statements, which have been flagged as harmful, abusive or offtopic, will be deleted, if positive collective consensus is reached.
- *Duplicate:* Statements which have been identified as a duplicates will be merged, if a positive collective consensus is reached.
- *Edit:* Proposals for updating already existing statements. If positive collective consensus is reached, these statements will be replaced with the proposed version.

[2] https://www.elastic.co/

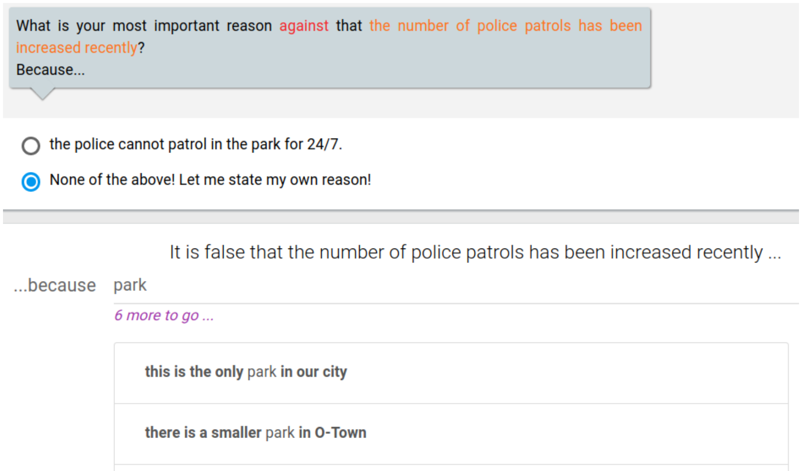


Figure 2.

- *Optimization Required:* Sometimes a user is not able or willing to provide an update for a statement that requires a revision. In this case the statement can be flagged and other users can provide an update while going through this queue. The update will then be submitted to the edit queue and the respective entry will be deleted from the optimization required queue once positive consensus is reached.
- *Split:* Even though we use framing, on rare occasions users include both premise and conclusion or multiple distinct premises in a single text contribution. This can be flagged by other users. If a positive collective consensus is reached, the flagged statement will split appropriately.

In order to motivate users to participate by providing statements or by taking part in the review system, they gain reputation by helpful actions and in order to deter them from abusing the system, they loose reputation if their actions are considered unhelpful. The actions that a user can take, in particular which review queue she can use, depends on the reputation of the user.

8. D-BAS: Implementing a Dialog-Based Online Argumentation System

We have developed an application for dialog-based online argumentation called *Dialog-Based Argumentation System* (D-BAS) which implements the ideas described above. It is available both as a web-based service[3] and as open source software[4].

D-BAS' backend is written in *Python 3* with usage of the *Pylons Pyramid web framework*. We use *nginx* as proxy, *uwsgi* as webserver and *Chameleon* as HTML template engine. Additionally we use *Node.js* with *Socket.IO* for asynchronous and bidirectional communications, e.g. notifications about specific events. D-BAS' data structure is managed by Pythons SQL toolkit *SQLAlchemy*. D-BAS' frontend is built upon a number of established technologies like *HTML*, *JavaScript* with *jQuery*, *Bootstrap* and *SASS*.

[3]https://dbas.cs.hhu.de
[4]https://github.com/hhucn/dbas

To allow future applications to use the functionality of D-BAS we implemented an *application programming interface* (API), which provides the possibility to access D-BAS' backend. This abstraction of the core argumentation functionality of D-BAS can then be used to enable dialog-based discussions in an arbitrary context.

Furthermore, it is possible to query the contents of the database in a very flexible way, where a developer can specify which data she wants to access. Most contents of our discussions are freely queryable except for private information, e.g. the user's password-hash. For this we implemented *GraphQL*, which allows flexible queries for the required data. Additionally, we offer *GraphiQL*[5] as an in-browser IDE for exploring GraphQL. Querying data or giving the transferred statements a structure is the main goal of the *Argument Interchange Format*, an universal format for the exchange of arguments by [16], but we decided to make it more flexible for the developers to query data from our database. All necessary information to use the API and more examples are available in the documentation of D-BAS[6].

9. Findings from a real-world online participation process based on D-BAS

D-BAS was used in a real-world online participation process where all students of our computer science department were invited to propose and discuss improvements to the computer science studies program. The main issue was how to deal with an increased number of students. The number of enrolled students has more than doubled in the past three years, leading to numerous problems such as overcrowded lectures.

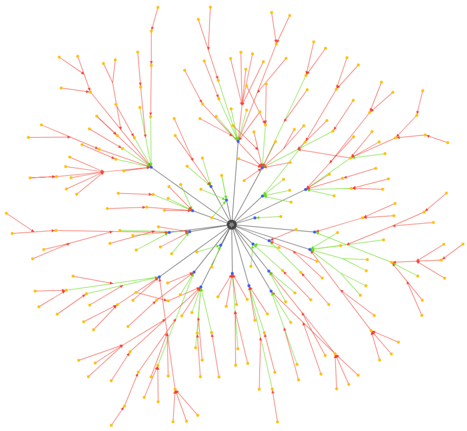
A full report on this process an its results can be found in [3]. Here, we only seek to answer the question: "Does D-BAS enable a large group of untrained participants to make proposals and discuss then in such a way, that the resulting WoR is reasonably well-formed and helpful for those that have to evaluate the proposals?"

The online participation process took place from may, 9^{th} until may, 28^{th} 2017 and all students of our computer science department were invited via e-mail. In total there were 318 unique visitors who added 22 positions and 255 statements. The resulting argumentation map is shown in Figure 3a. The typical depth of a sequence of arguments varies between three and four. This clearly shows that there was (time-shifted) interaction between the participants.

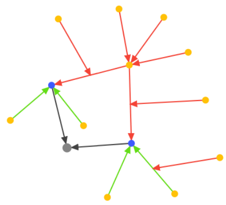
In order to allow others to analyze the discussion, we summarized the main facts online[7] and we offer a dump of our database, which is licensed under the Creative Commons License CC BY-NC-SA[8].

As a first step we investigated the quality of the resulting argumentaion by taking a look at the proposed positions. The students added 22 positions to the argumentation, where each position lead to further reactions, indicating that they were of interest to others. Furthermore, no position was a duplicate of another position and all proposed positions where reasonable. While it is not possible to prove that no other means of online argumentation might lead to more or better positions, the absolute number indicates that the argumentation was extremely successful at gathering meaningful positions.

[5] https://dbas.cs.hhu.de/graphiql
[6] https://dbas.cs.hhu.de/docs/api/v2.html
[7] https://dbas.cs.hhu.de/fieldexperiment
[8] https://creativecommons.org/licenses/by-nc-sa/3.0/



(a) Argumentation graph created by participants in D-BAS. The grey dot is the root of the discussion, the blue dots are positions and the yellow dots are statements that are not positions. Green arrows denote supporting arguments and red arrows denote attacking arguments.



(b) Connected subgraph during a discussion.

Second, we examined how the participants interacted during the online argumentation. The discussion consists of 265 statements in total. In order to examine the participant's interactivity, it is important to understand how the results of the argumentation look like. Essentially, each position is the start of a sub-graph of arguments. Since statements can be reused, the sub-graphs of the positions are interconnected, as shown in Figure 3b. The size of the subgraphs was between 2 and 44 with an average of 13. This shows that for all proposals there was a significant exchange of arguments.

Third, we analyzed the selected feedback options. Users selected 200 undermines, 44 supports, 137 undercuts, 56 rebuts; 19 times they wanted to see another attacking argument and 104 times they went a step back. We manually investigated if those reactions were used appropriately, that is, if the reaction made sense in relation to the argument it was a reaction to. This holds true for every single reaction. This is surprising, since at least the undercut is a challenging type of reaction. While we were very pleased with this result, it should be noted that the participants were all computer science students. It is not certain that this result would remain unchanged with a different set of participants.

Summarizing, the field experiment indicates that it is possible to lead a high quality online argumentation by using dialog-based online argumentation. It demonstrates in a real-world setting that participants with no background in formal argumentation are able to collectively argue about a topic in such a way that the resulting formal argumentation map is reasonable and non degenerated.

10. Conclusion

In this paper we have presented D-BAS, a system for dialog-based online argumentation. We have identified and solved three main challenges: providing feedback on existing arguments, selecting the next argument that should be presented to the user and incorporating user input. The resulting system was fully implemented as an open source, web-based service with a well-defined API that can be used by other applications. Further, in a real world deployment we have shown that untrained participants of an e-participation process are able to use D-BAS and that the resulting WoR is reasonably well-formed.

Acknowledgements

This work was done in the context of the PhD programme "Online Participation"[9], funded by the Ministry of Innovation, Science and Research in North Rhine Westphalia, Germany. We thank Björn Ebbinghaus and Marc Feger for their assistance.

References

[1] M. Klein, "Using Metrics to Enable Large-Scale Deliberation," in *Collective intelligence in organizations: A workshop of the ACM Group 2010 Conference*, 2010, pp. 103–233.

[2] T. Krauthoff, M. Baurmann, G. Betz, and M. Mauve, "Dialog-Based Online Argumentation," in *Proceedings of the 2016 Conference on Computational Models of Argument*. iOS Press, 2016.

[3] T. Krauthoff, C. Meter, and M. Mauve, "Dialog-Based Online Argumentation: Findings from a Field Experiment," in *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence*, vol. 2012. CEUR-WS, 2017, pp. 85–99.

[4] L. Iandoli, I. Quinto, P. Spada, M. Klein, and R. Calabretta, "Supporting argumentation in online political debate: Evidence from an experiment of collective deliberation," *New Media & Society*, 2017.

[5] M. Van Alstyne, E. Brynjolfsson *et al.*, "Electronic Communities: Global Village or Cyberbalkans?" in *Proceedings of the 17th International Conference on Information Systems. New York: Wiley*, 1996.

[6] C. R. Sunstein, "The law of group polarization," *Journal of political philosophy*, vol. 10, no. 2, pp. 175–195, 2002.

[7] M. Klein, "Enabling large-scale deliberation using attention-mediation metrics," *Computer Supported Cooperative Work (CSCW)*, vol. 21, no. 4-5, pp. 449–473, 2012.

[8] T. Kriplean, J. Morgan, D. Freelon, A. Borning, and L. Bennett, "Supporting Reflective Public Thought with ConsiderIt," in *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*. ACM Press, 2012, pp. 265–274.

[9] T. F. Gordon and D. Walton, "The Carneades Argumentation Framework – Using Presumptions and Exceptions to Model Critical Questions," in *6th computational models of natural argument workshop (CMNA), European conference on artificial intelligence (ECAI), Italy*, vol. 6, 2006, pp. 5–13.

[10] M. Klein and L. Iandoli, "Supporting Collaborative Deliberation Using a Large-Scale Argumentation System: The MIT Collaboratorium," 2008.

[11] S. Wells, "Supporting Argumentation Schemes in Argumentative Dialogue Games," *Studies in Logic, Grammar and Rhetoric*, vol. 36, no. 1, pp. 171–191, 2014.

[12] T. Bench-Capon, K. Atkinson, and A. Wyner, "Using Argumentation to Structure E-Participation in Policy Making," *Transactions on Large-Scale Data-and Knowledge-Centered Systems XVIII*, vol. 8980, pp. 1–29, 2015.

[13] T. F. Gordon, "Structured Consultation with Argument Graphs," *From Knowledge Representation to Argumentation in AI. A Festschrift in Honour of Trevor Bench-Capon on the Occasion of his 60th Birthday*, pp. 115–133, 2013.

[14] F. Bex, J. Lawrence, M. Snaith, and C. Reed, "Implementing the Argument Web." *Communications of the ACM*, vol. 56, no. 10, pp. 66–73, 2013.

[15] S. Wells and C. Reed, "MAgtALO: an Agent-Based system for persuasive online interaction," in *AISB 2008 Convention Communication, Interaction and Social Intelligence*, vol. 1. Citeseer, 2008, p. 29.

[16] C. Chesñevar, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South, G. Vreeswijk, and S. Willmott, "Towards an Argument Interchange Format," *The Knowledge Engineering Review*, vol. 21, no. 04, pp. 293–316, 2007.

[9] `http://fortschrittskolleg.de/en`

3.2 Dialog-Based Online Argumentation: Findings from a Field Experiment

In this section we present the findings from a first field experiment with DBOA from Krauthoff et al. 2017:

Tobias Krauthoff, Christian Meter and Martin Mauve.

"Dialog-Based Online Argumentation: Findings from a Field Experiment"

In: *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence (AIˆ3 2017)*, *AI*IA Series*, pages 85–99, CEUR Workshop Proceedings Volume 2012.
Acceptance Rate: ∼65%

Summary

Since one of our key goals is to build people-centric software, it is necessary to involve (especially untrained) users into the development process. In this case, we started a field study with students from the Heinrich-Heine-University Düsseldorf to participate in a discussion about ideas to improve the computer-science studies. All computer-science students and employees from our faculty were invited to participate. No financial incentive was given and it was completely free to join the discussions. The office of the dean announced that they would comment on the suggestions after the discussion to verify the feasibility of the proposed actions.

The first chapters describe DBOA and D-BAS, because the full paper of D-BAS (see Section 3.1) was still under review.

This study was the first real-world test with untrained users. One key goal was to test that the dialog-based approach works with a broad user range and produces structured, machine-readable data. During three weeks, 318 unique persons visited our argumentation system and participated in or followed the discussion. In total, 255 statements and 22 positions were provided by the participants and the DMS was heavily used. We have found that users can be divided in several groups, which each follow a specific pattern, e.g. not commenting, but moderating the whole discussion through the moderation system, or one-off users giving one argument and then never joining the discussion again.

Summarizing, the field study went well. We automatically constructed a huge argumentation graph, had no technical issues and learned a lot from the participants.

Afterwards, we invited all participants to fill out a survey concerning their experiences with D-BAS. Our software was in general accepted and understandably usable. The DMS was positively evaluated as well and heavily used to rework the arguments from other participants.

Personal Contribution

The contributing author, Christian Meter, jointly prepared the experiment with Tobias Schröder (né Krauthoff) and was involved in the development process of stabilizing and hardening the argumentation core of D-BAS to achieve its production-readiness. Presenting the overall statistics of a discussion to provide live numbers on proposed statements and introducing a DMS were Christians ideas. Tobias then implemented these ideas.

Christian wrote about the decentralized moderation tool (Section 5) and parts of Section 3. Martin Mauve supervised the paper and provided an introduction. Tobias wrote the remaining parts of this paper, including the evaluation of the survey questions and processing the statistics of proposed statements and user behavior. All three authors worked together on survey questions for the participants.

Importance and Impact on this Thesis

To fulfill the goal of building people-centric software, it is necessary to conduct field tests with untrained users. This paper confirmed that the dialog-based approach has practical impact on the way people are discussing and improves discussion culture in this academical context. A huge benefit is the structured data, which is collected through the dialog of our systems. We reached a *high number of valid arguments* and their relations, which is now automatically analyzable by algorithms for further processing. The substitution of text and argumentation response options was well-chosen and implemented, as it could be read from the survey results.

It also showed that we still have usability problems, e.g. when it comes to joining the discussions. Participants often did not use the starting point via positions, but chose the argumentation graph instead to find the last point they were discussing. Also, some participants did not write their comments in the proposed fields, so instead of having one or more premises, they provided more than one argument in the premise-field.

The field experiment indicates that the DBOA approach and the democratic moderation with the DMS work. Further research can now be done based on this study, taking into account that most problems we encountered seemed to be usability-difficulties.

Dialog-Based Online Argumentation: Findings from a Field Experiment

Tobias Krauthoff, Christian Meter, and Martin Mauve

Department of Computer Science, University of Düsseldorf, Germany
{krauthoff,meter,mauve}@cs.uni-duesseldorf.de

Abstract. In this paper we report on the results of a field experiment where more than 300 participants used dialog-based online argumentation. The participants were computer science students discussing how to improve the computer science course of studies. At the beginning of the argumentation the participants were informed that the results would be carefully considered by the computer science department in order to revise the course of studies. Thus this was a real-world experiment and not an artificial lab setting.
Over the course of two weeks the online argumentation received 255 user-submitted statements, leading to 235 arguments. After the argumentation was concluded we carefully analyzed the resulting content and asked the participants to answer a questionnaire. Our findings indicate that dialog-based online argumentation can result in a high-quality exchange of arguments without the need of anyone involved being an expert on formal argumentation. Furthermore we identified several areas where dialog-based online argumentation and our specific implementation could be improved significantly.

Keywords: dialog-based, argumentation, field experiment, large-scale discussion

1 Introduction

Dialog-based online-argumentation is an online argumentation scheme, where participants are guided through the arguments provided by other users, so that they perform a time shifted dialog with those that have participated before them. It does not require any prior knowledge or training from the users and avoids the shortcomings of forum-based systems, in particular balkanization and lack of scalability. Dialog-based online-argumentation is driven by a formal data structure capturing the full complexity of argumentation. The user interaction, however, has the structure of a regular dialog as it is performed in everyday life.

We have introduced the idea of dialog-based online-argumentation in [9]. In that paper we discussed the challenges and potential solutions required to build a dialog-based online-argumentation system and presented a first prototype, called *Dialog-Based Argumentation System* (D-BAS)[1], which is available on *GitHub* as

[1] https://dbas.cs.uni-duesseldorf.de/

open source software[2]. Since then, we have improved and extended D-BAS into a fully fledged system for dialog-based online argumentation, so that we are now able to leave the lab and lab-experiments behind and instead deploy and evaluate D-BAS in real world settings.

In this paper we describe the findings from a real world use of dialog-based online argumentation, where all students of our computer science department were invited to propose and discuss improvements to the computer science studies program. In particular this includes an analysis of how the users participated in the discussion, an investigation of the user-based review system provided by D-BAS, information on the resulting arguments and their structure as well as information from a user survey. Furthermore we provide free access to the resulting argumentation data both in the native language of the argumentation (German) and an English translation. Both language versions are downloadable[3] as data sets for further study and are included in the live version of D-BAS, so that anyone interested can review the discussion in detail.

This paper is structured as follows. In Sec. 2 we give a brief overview of related work in the area of online argumentation. The general idea of dialog-based online argumentation and its implementation in D-BAS is summarized in Sec. 3. Section 4 describes the setting of the field experiment. Section 5 has a closer look at the peer-based review system and how it was used by the participants of the discussion. The quality of the resulting online-argumentation is investigated in Sec. 6. The results from a survey taken by the participants of the discussion is presented Sec. 7. We conclude the paper with a brief summary and an outlook to future work in Sec. 8.

2 Related Work

Tools for asynchronous online-discussion can be separated into forum-based approaches, pro and contra lists and tools for argument mapping. Although forum-based approaches received quite a lot of criticism in the past [7], it is, by far the most commonly used approach to support online argumentation in practice.

It has been suggested to use online pro and contra lists to aid collective decision making processes like *ConsiderIt* [10]. These lists work very well for evaluating a given proposal, but they are not suitable to deal with more general positions and alternatives since they do not support the exchange of arguments and counter arguments.

Online systems for argument mapping enable participants to structure their arguments and the relation between them in an argument map. While those systems do avoid the shortcomings of forum-based approaches, they require the users to become familiar with their notations and the semantics of formal argumentation. Examples are *Carneades* [4, 3], *Deliberatorium* [8] and *ArguNet* [11]. Therefore, in practice, they are used by skilled users, who are familiar with logic

[2] https://github.com/hhucn/dbas
[3] https://dbas.cs.uni-duesseldorf.de/static/data/fieldtest_05_2017.tar.bz2

of argumentation rather than by average participants that want to take part in an online argumentation.

The idea of engaging in a formalized dialog to exchange arguments is used by dialog games, where participants follow a set of rules to react to each others statements [12]. In contrast to our work, dialog games look at the real-time interaction between users in order to learn something about a subject at hand. They do not seek to provide better instruments for online argumentation.

In addition to the main classes of ideas presented above, there is an individual system that is related to our work: Arvina [1]. Arvina allows a user to conduct a dialog between robots and humans. As a basis, it uses an existing discussion specified in a formal language [2] where the positions and arguments of some real-world persons are marked. A robot can use this information to argue with human participants. The participants can query the robots and each other. In contrast to the system we envision, *Arvina* is driven by the questions of the users. Thus there is no need for the users to react to replies from the system by providing their own arguments.

3 Dialog-Based Argumentation System

The goal of dialog-based online argumentation is to enable any user to participate efficiently in a large-scale online argumentation. At the same time it seeks to avoid, or at the very least reduce, the problems that occur in unstructured online argumentation such as a high level of redundancy, balkanization, and logical fallacies. The result of dialog- based online argumentation is a set of user-provided statements, their interrelation and the opinion of the participants on both statements and relations between statements.

In the following, we briefly describe terms that will be used to explain the main aspects of dialog-based online argumentation. Based on these terms, we then introduce the main concepts of dialog-based online argumentation.

Each discussion is a set of *statements*, which are the most basic primitives used in an online discussion. The negation of a statement is itself a statement. Individual participants might consider a given statement to be true or false. A *position* is a prescriptive statement, i.e., a statement which recommends or demands that a certain action can be taken. Furthermore we need to distinguish between first-order and second-order *arguments*. A first-order argument consists out of a premise group — a set of at least one statement — and a conclusion, i.e. a statement. Both are connected by an inference, which is either supporting or attacking, so that the premise group is a reason for or against the conclusion. A second-order argument has the same kind of premise group, but the conclusion is the inference of an argument. With this we can argue about the validity of another reason-relation. Together, the arguments of a debate form a (partially connected) *web of reasons*.

The core idea of dialog-based online argumentation is a loop consisting of three steps: (1) presenting a single argument; (2) gather feedback from the user based on a list of alternatives and (3) the system selecting the next argument that is shown to the user based on the response and, possibly, the data gathered from the responses of other participants [9]. In this way the user and the system perform a *dialog* where the system selects arguments that are likely to be of interest to the user and where the user provides feedback on those arguments.

The first thing that the system needs to do when a new user wants to participate in the online discussion is to choose an initial argument. This is challenging since the system has no information on the user, yet. One fairly straightforward solution is to simply ask the participant for an initial position she is interested in (see Fig. 1). After she has chosen or provided her position, she is asked to select or provide a statement explaining her choice (see Fig. 2 and Fig. 3). This statement is used as the premise, whereas the position forms the conclusion.

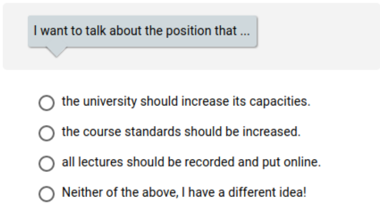


Fig. 1. Choosing an initial position.

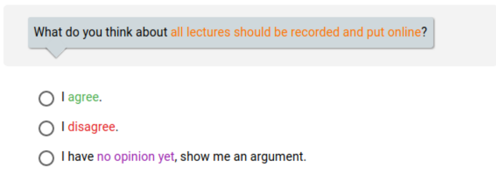


Fig. 2. Choosing attitude towards a position.

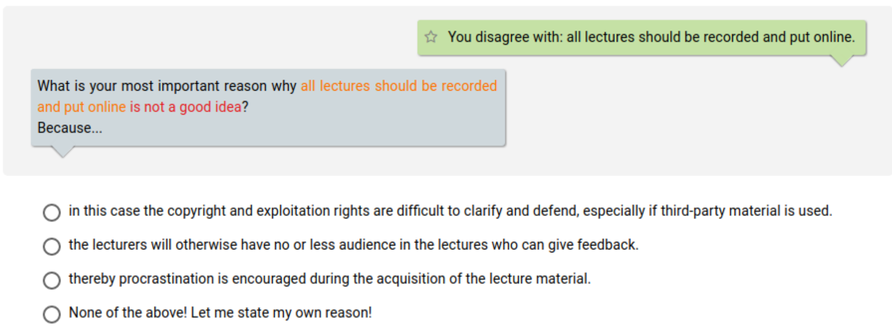


Fig. 3. Selecting a premise for the initial argument.

Once a user is confronted with an argument (see Fig. 4), she can provide feedback on the argument. The options have to be usable by unskilled participants, but also have to be logically correct. We propose the following: (1) Reject the premise. (2) Accept the premise and, as a consequence, the conclusion. (3) Accept the premise but disagree that this leads to accepting the conclusion. (4) Accept the premise but state that there is a stronger argument that leads to rejecting the conclusion. (5) Do not care about the argument. Depending on the choice of the user, she can provide a statement supporting her feedback on the presented argument. This may be taken from a list of existing statements (see Fig. 5) or she may enter a new one (see Fig. 6). While entering a new statement, the system scans for similar statements that have already been provided by other users and displays them in a ranked list. In this way it is easy to reuse existing statements while avoiding duplication of statements in the web of reasons. Any new statement added by the user will be inserted in the web of reasons.

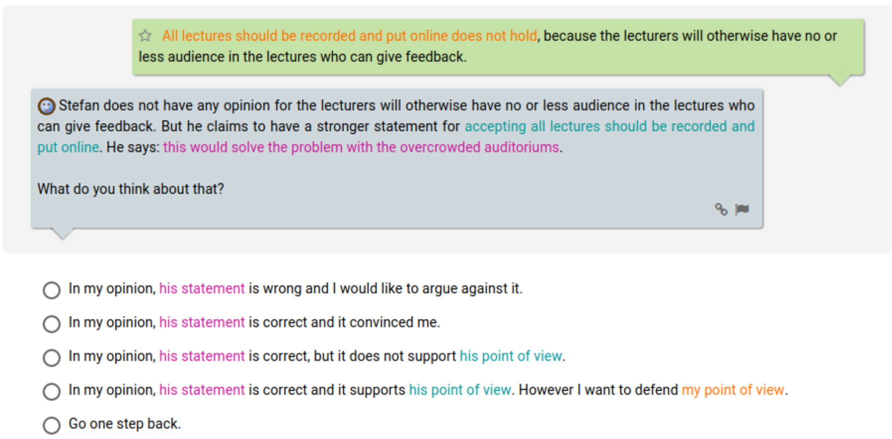


Fig. 4. Challenging the user's argument and getting feedback from the user.

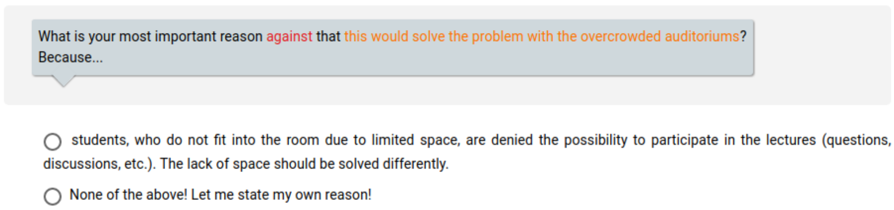


Fig. 5. Justification of the opinion in D-BAS.

4 Setting of the Field Experiment

The field experiment, we report about in this paper, took place at the computer science department of the Heinrich-Heine-University Düsseldorf. It targeted a

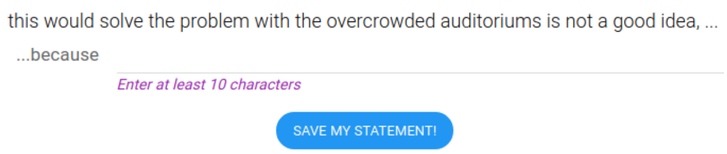


Fig. 6. User interface for entering a new statement.

topic that was relevant to the students of the department: how to deal with the increased number of students. The number of students has more than doubled in the past three years leading to numerous problems such as overcrowded lectures and a lack of places where students could sit down and study either in groups or by themselves. In order to avoid that participants are confronted with an "empty" system, we initialized D-BAS with two positions as well as two pro and two contra statements for each of those positions.

The students of the department were then invited via mail on behalf of the dean of the faculty of mathematics and natural sciences on May, 9^{th} of 2017. Furthermore the teaching assistants of the department were invited, as well. The participants were asked to discuss how the course of study can be improved and how the problems caused by the large number of students can be reduced. The discussion was open until May, 28^{th} of 2017. In total, there were 318 unique visitors and 47 users logged in to the system. Logging in is required to enter a new statement while conducting a dialog with the systems can be done anonymously. Out of the 47 users who logged in 11 were female and 36 were male. This roughly reflects the distribution of male and female students in the department. In total the participants added 22 positions and 255 statements (including the 22 positions). The resulting argumentation map is shown in Fig. 7[4].

In order to allow others to analyze the discussion, it is available for download[5] as a dump of a *PostgreSQL* database and is licensed under the Creative Commons License *CC BY-NC-SA*[6]. The archive contains three versions: the original dataset of the discussion in German, a dataset which includes some corrections (those corrections are described in detail in Sec. 6) in German and a translation of the corrected dataset translated to English.

5 Decentralized Moderation

Dialog-based Online Argumentation relies on statements provided by the users in order to construct arguments that are then used in the dialog with other participants. In order to encourage users to provide well-formed statements,

[4] https://dbas.cs.uni-duesseldorf.de/discuss/improve-the-course-of-computer-science-studies#graph
[5] https://dbas.cs.uni-duesseldorf.de/static/data/fieldtest_05_2017.tar.bz2
[6] https://creativecommons.org/licenses/by-nc-sa/3.0/

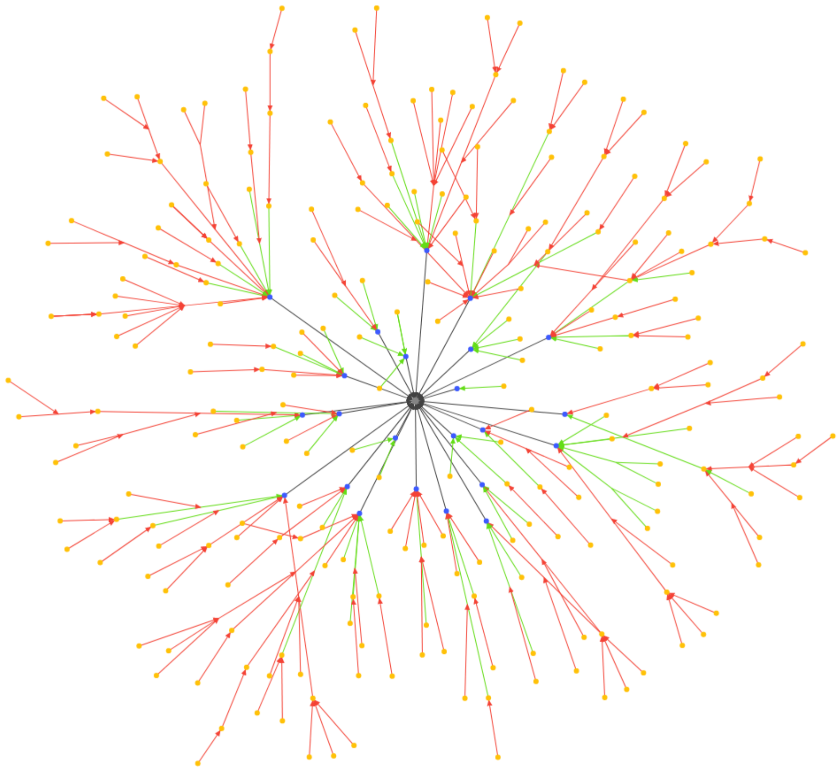


Fig. 7. Argumentation graph created by participants in D-BAS. The grey dot is the root of the discussion, the blue dots are positions and the yellow dots are statements that are not positions. Green arrows denote supporting arguments and red arrows denote attacking arguments

D-BAS provides a specific context when statements are entered, for example “Lectures should be recorded and released on a streaming platform because ...”. This will usually nudge the user towards entering a statement that completes the sentence in a meaningful way. Of course, this cannot completely prevent errors or malicious behaviour. It is therefore necessary to have a means for moderating the content provided by the users.

This could have been done by providing an interface where dedicated moderators would be able to alter or delete the statements provided by the regular users. If those moderators are skilled in argumentation and familiar with D-BAS, they could even make sure that statements are well formed for the use in D-BAS. We did not chose to take this approach. Instead we wanted to see if a decentralized moderation by the (untrained) participants themselves could work as well. This would be an important finding, since it would show that dialog-based online argumentation can take place and lead to a complex formal argumentation structure without anyone involved knowing anything about formal argumentation.

The decentralized moderation system implemented in D-BAS has been inspired by *Stack Overflow*[7] and works as follows. Every participant can flag content. She can either provide an improved version of the flagged content or simply report it as “The statement needs to be revised” or “This statement is off-topic or irrelevant” or “This statement is harmful or abusive” or “This statement is a duplicate”. Flagged content is not changed immediately. Instead it is entered into one out of several review queues, depending on how it was flagged. For example if a statement is flagged as harmful or abusive it is entered in the “Delete” review queue. Other users can go through those queues and either vote on the action to be taken or provide an alternative version of the flagged statement. Once a sufficiently clear-cut collective opinion has been reached, the appropriate action is taken, e.g. the statement might be replaced or deleted or the flagging might be discarded. The review queues maintained by D-BAS are as follows:

Delete: This queue contains statements, which have been flagged as off topic, irrelevant, harmful or abusive. If positive collective consensus is reached, this statement will be deleted.

Edit: This queue contains proposals where users have submitted and revised version of an existing statement. If positive collective consensus is reached, the old statement will be replaced by the new one.

Duplicate: It may happen that two separate statements are provided by users even though those statements have the same meaning. In this case it would make the argumentation more straight forward if those statements were merged. Those duplicate statements can be reported in the following way: one statement is marked as a basis and then another statement is selected as the duplicate. If positive collective consensus is reached, the duplicate will be deleted and the original statement will replace it.

Optimization: Finally, statements may be flagged because they need to be revised. Users going through the optimization queue can provide an alternative version of a statement from the optimization queue. This revision is then submitted to the edit queue for review.

In order to motivate users to participate by providing statements or by taking part in the review system, they gain reputation by helpful actions and in order to deter them from abusing the system, they loose reputation if their actions are considered unhelpful. The actions that a user can take in D-BAS, in particular which review queue he can use, depends on the reputation of the user.

During the discussion at hand, 47 statements were flagged: no deletes, 25 edits, 5 duplicates and 17 requests for optimization. Figure 8 shows the results

[7] https://stackoverflow.com/review

of the voting on the flagged statements. This excludes requests for optimization since those will not result in a vote but in an updated statement which is then submitted to the edit queue. The vast majority of flagged statements is decided upon unanimously with three votes in favour of positive consensus. Only very few decisions required more than three votes to reach a decision, whereby the limit is five. The two instances marked in red were not decided upon at the end of the discussion, since they have not received a sufficient number of votes. This happened since they were flagged close to the end of the discussion.

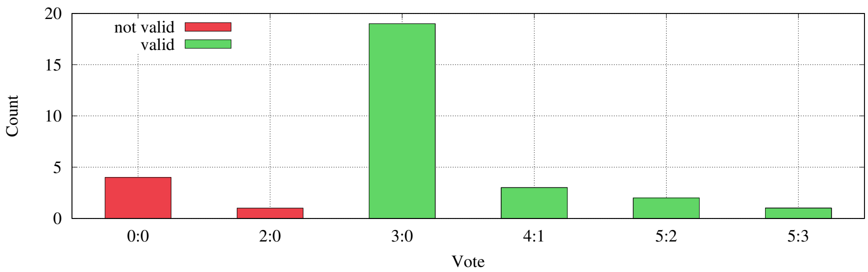


Fig. 8. Overview of voting in the D-BAS review system.

In the discussion, positive consensus was reached in every single case where any consensus was reached at all: all actions proposed by the user flagging the content were taken and all proposals for updating statements where accepted. We checked manually, if those decisions were plausible and found that this is, in fact, the case. All statements flagged as duplicates were true duplicates and every single edit corrected at least some mistake in the original statement. Also, there were no duplicates remaining that have not been flagged. However, some of the edits introduced new (mostly spelling) errors. This might also explain the non unanimous votes.

We were interested in how participation was distributed among the participants of the discussion in the review system. Figure 9 shows the share of each user for contributing statements, flagging statements and actions taking in the review system. It is quite obvious that for each type of action there are some power users. However, those are not the same across all action types. It seems that distinct users enjoy different aspects of contribution to the discussion.

Clearly, the discussion took place in a benign setting. A more controversial topic discussed by a less homogeneous group might stress the distributed review system to a significantly larger extent. However, what our findings clearly show, is that regular users will participate in the review system and that they are able to collectively improve the quality of individual statements and the overall discussion.

From observing the discussion we also learned, that there should be two more review queues. One for statements that should be split into several distinct

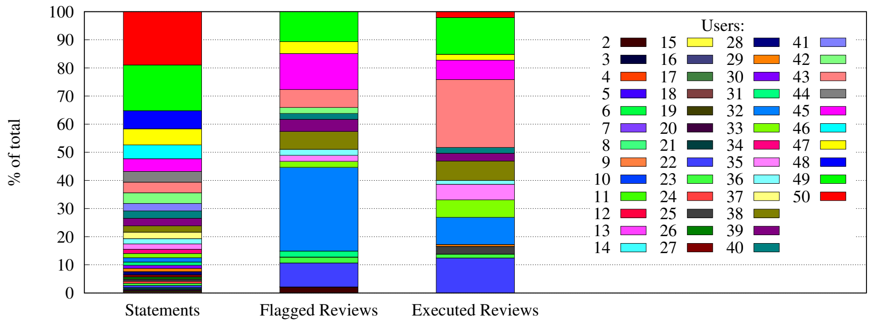


Fig. 9. Distribution of the users activity in D-BAS.

statements. This would come in handy if an inexperienced user includes both premise and conclusion or multiple distinct premises in a single text contribution. Another one for handling the opposite case, i.e., restoring a statement that has incorrectly been split into multiple parts. The specific observations that led us to those conclusions will be discussed in more detail in the following section.

6 Quality of the Argumentation

One key question we wanted to answer with the field experiment was whether dialog-based online argumentation works and can, in fact, lead to a good online argumentation. Obviously, there is no simple metric that one could use to decide whether this is the case or not. However, it is possible to investigate individual characteristics of the argumentation that, taken together, provide a strong hint regarding its quality.

First, we take a look at the positions that were proposed by the participants. Positions are statements that can be executed. In this specific argumentation they represent ideas on how the computer science studies program can be improved. Altogether the participants added 22 positions to the argumentation. As mentioned above, additionally, two positions were provided by us at the start of the field test. All of the positions added by the participants are meaningful in the sense that they are actions that could potentially have an impact on the quality of the studies program. They all led to further reactions by other participants, indicating that they were of interest to others. Furthermore, there were no duplicate positions. This is an important prerequisite for scalability. While it is not possible to prove that no other means of online argumentation might lead to more or better positions, the absolute number indicates that the argumentation was extremely successful at gathering meaningful positions.

Next, we investigate how interactive the online argumentation was. The argumentation consists of 265 statements, including the 24 positions. In order to investigate interactivity, it is important to understand how the results of the argumentation look like. Essentially, each position is the start of a sub-graph of arguments. Since statements can be reused, the sub-graphs of the positions are interconnected. From the perspective of the individual positions they overlap. An example for two overlapping subgraphs from the discussion is shown in Fig. 10[8].

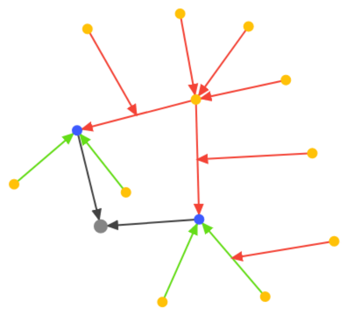


Fig. 10. Connected subgraph during a discussion.

In order to determine the interactivity of the argumentation, we can now look at the number of statements that are directly or indirectly connected to each position. Furthermore we can investigate the maximum length of chains of arguments that are connected to each position.

Both the number of statements related to each position and the length of argument chains for each position are shown in Fig. 11. Most positions attracted more than ten arguments with the maximum at around 45 arguments for one position. Also, each position led to an average argument chain of length three or four. This clearly shows that this was a very interactive argumentation. Furthermore, the argumentation does not contain any (obvious) duplicate statements. Again, this is an important prerequisite for scalability. However, this is due to the review system and not an inherent attribute of dialog-based online argumentation: the participants themselves detected and removed five duplicated statements over the course of the argumentation using the review system.

One important aspect regarding the quality of an argumentation is whether the participants are able to react to arguments of others in an appropriate way. Given an argument consisting of a set of premises and a conclusion, D-BAS allows for the reactions described in Sec. 3 and shown in Fig. 4. Based on each participants history, recorded by Piwik[9], we analyzed the selected feedback options. During the field test users have selected 200 undermines, 44 supports, 137 undercuts, 56 rebuts, 19 times they wanted to see another attacking argument and 104 times they just wanted to go back. We manually investigated, if those reactions were used appropriately, that is, if the resulting argument makes sense

[8] https://dbas.cs.uni-duesseldorf.de/discuss/improve-the-course-of-computer-science-studies/attitude/454#graph

[9] Piwik is an open-source analytics platform: https://piwik.org/.

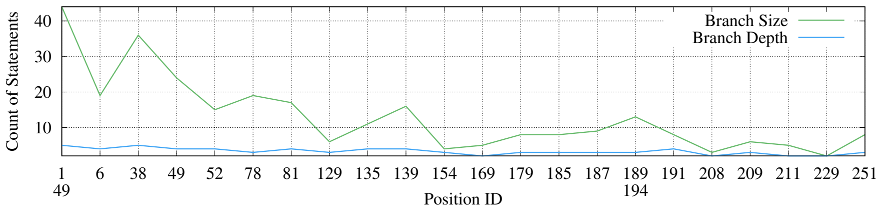


Fig. 11. Size of sub-graphs and length of argument chains for each position.

in relation to the argument it was a reaction to. This holds true for every single reaction. This is surprising since at least the undercut is a challenging type of reaction. While we were very pleased with this result, it should be noted that the participants were all computer science students. It is not certain that this result would remain unchanged with a different set of participants.

So far all aspects of the argumentation indicate that dialog-based online argumentation and the D-BAS implementation indeed support high quality online-argumentations. However, as we will show next, there have also been some problems that we could observe. All of them are caused by the current D-BAS implementation and all of them can be avoided in the future by adapting the implementation accordingly.

During the experiment we had to intervene three times in order to split a single contribution of a user into several separate statements. In each of these cases we feared that not intervening would lead to follow-up problems when other users would try to react to the contribution of the user.

The first two cases occurred while the user was entering a position. Instead of just entering a position the user also provided a justification for the position. This problem happened, because the respective participant did not know that right after entering a position she would be asked for a justification for the position. This problem occurred only twice, because as soon as one had used D-BAS for a very brief time, it would become obvious that one should enter only the position at this time. In the future we will prevent this problem by merging the two steps of providing a position and its justification so that a user immediately realizes that she can provide the justification for the position in a separate entry field.

In the third case a user provided several separate premises in one contribution. This is a problem, because it would then not be possible for other participants to address each premiss individually. Again, after getting familiar with D-BAS, it would be obvious that one should provide only separate statements. Since we can not completely prevent this from happening, however, we will add an option to the review system that would allow other participants to break down a contribution like this into separate statements. Since this functionality was not present in the version of D-BAS we used in the field experiment, we manually split the contribution.

Additionally, we discovered that one feature of our user interface was misleading, if the user did not pay close attention: we assumed that the usage of the keyword "and" in a statement would often mean that the user tried to connect multiple statements that would better be represented as separate statements. Whenever a participant used "and", D-BAS therefore explicitly asked if it should split the statement. If the user, at this point, did not choose the correct answer, a single statement that included "and" would be split in two meaningless fractions of a statement. While in the vast majority of cases where "and" was used, the participant choose the right option, there were six occurrences were they did not. We did not correct those issues while the discussion was under way, since they did not significantly hamper the discussion itself. However, in order to make the resulting data more accessible, we corrected them later on. For transparency reasons, we also kept the original data set.

In order to avoid this problem in the future, we will simply allow users to recombine those statements using the review system. This will solve this issue, since the problem is really obvious as soon as D-BAS splits the statements.

Summarizing, while there have been minor problems caused by the current version of D-BAS, the field experiment clearly shows that it is possible to lead a high quality and redundancy free online argumentation by using dialog-based online argumentation and its implementation, D-BAS. In particular, it demonstrates in a real-world setting that participants with no background in formal argumentation are able to collectively argue about a topic in such a way that the resulting formal argumentation map is correct and very comprehensive.

7 User Feedback

As a follow-up to the online discussion, we invited all participants to take part in a survey about D-BAS. As an online survey tool we used *Unipark*[10].

Figure 12 shows the attitude of the participants towards key statements regarding D-BAS. For each line, the number of participants that answered the question is given. Clearly, the participants that have answered those questions do have a positive attitude towards D-BAS. In particular, they seem to like the general approach taken by D-BAS and they would use D-BAS again. It is also noteworthy, that for every single statement the average attitude is at or above neutral.

We were also interested in the attributes that users would associate with D-BAS. As a means to investigate this, we used bipolar word pairs. The result of this is shown in Fig. 13. Again, the results show that users participating in the survey assign quite positive attributes to D-BAS. However, they also indicate, that there are areas where it could be improved. In particular this holds true for the orientation that users have during an ongoing dialog (clear vs. confusing and unpredictable vs. predictable). We will address this in future versions of D-BAS by displaying a miniature version of (a part of) the argumentation graph during

[10] http://www.unipark.com/en/

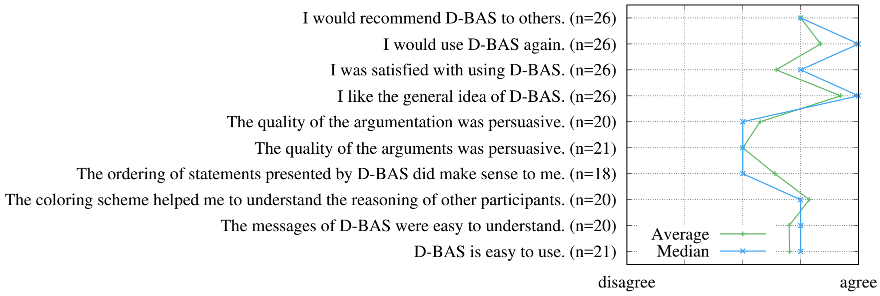


Fig. 12. Users evaluation of usability questions, based on *SUMI* [6].

the dialog. This should help the user to keep track of her position in the overall argumentation.

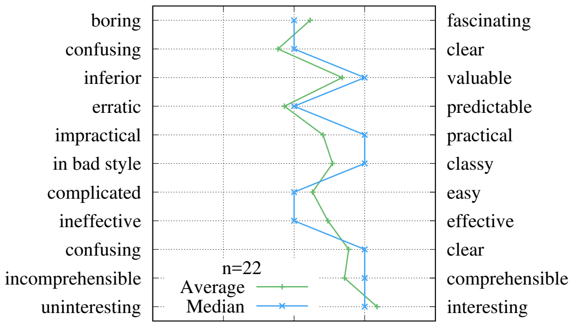


Fig. 13. Users evaluation of bipolar word pairs, based on *AttracDiff* [5].

8 Conclusion

In this paper we reported on the findings of a first field experiment using dialog-based online argumentation in a real world setting. The experiment confirmed, that this argumentation scheme is accessible by untrained participants and can result in a high-quality argumentation.

While the experiment provided us with a lot of information it is limited by the fact that this was only a single experiment with a very specific set of participants. In the future we will revise D-BAS according to the ideas presented here and make it available as a web-based service that anyone can use to host their online argumentation. Our goal is to collect the data from a large number of argumentations so that we can then investigate dialog-based online argumentation on a much larger scale.

Acknowledgements

This work was done in the context of the graduate school on online participation, funded by the ministry of innovation, science and research in North Rhine Westphalia, Germany. We thank Teresa Uebber for her assistance with the implementation of the argumentation graph.

References

[1] T. Bench-Capon, K. Atkinson, and A. Wyner. Using Argumentation to Structure E-Participation in Policy Making. *Transactions on Large-Scale Data-and Knowledge-Centered Systems XVIII*, 8980:1–29, 2015.

[2] F. Bex, J. Lawrence, and C. Reed. Generalising argument dialogue with the Dialogue Game Execution Platform. In *Computational Models of Argument: Proceedings of COMMA*, pages 141–152, 2014.

[3] T. F. Gordon. Carneades - tools for argument (re)construction, evaluation, mapping and interchange. http://carneades.github.io/, 2015. [Online, Last access 2017-06-27].

[4] T. F. Gordon and D. Walton. The Carneades Argumentation Framework – Using Presumptions and Exceptions to Model Critical Questions. In *6th computational models of natural argument workshop (CMNA), European conference on artificial intelligence (ECAI), Italy*, volume 6, pages 5–13, 2006.

[5] M. Hassenzahl. The interplay of beauty, goodness, and usability in interactive products. *Human-computer interaction*, 19(4):319–349, 2004.

[6] J. Kirakowski and M. Corbett. Sumi: The software usability measurement inventory. *British journal of educational technology*, 24(3):210–212, 1993.

[7] M. Klein. Using Metrics to Enable Large-Scale Deliberation. In *Collective intelligence in organizations: A workshop of the ACM Group 2010 Conference*, pages 103–233, 2010.

[8] M. Klein and L. Iandoli. Supporting Collaborative Deliberation Using a Large-Scale Argumentation System: The MIT Collaboratorium, 2008.

[9] T. Krauthoff, M. Baurmann, G. Betz, and M. Mauve. Dialog-Based Online Argumentation. *Proceedings of the 2016 conference on Computational Models of Argument (COMMA 2016)*, 2016.

[10] T. Kriplean, J. Morgan, D. Freelon, A. Borning, and L. Bennett. Supporting Reflective Public Thought with ConsiderIt. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pages 265–274. ACM Press, 2012.

[11] D. C. Schneider, C. Voigt, and G. Betz. Argunet – A software tool for collaborative argumentation analysis and research, 2006.

[12] S. Wells. Supporting Argumentation Schemes in Argumentative Dialogue Games. *Studies in Logic, Grammar and Rhetoric*, 36(1):171–191, 2014.

3.3 Various Efforts of Enhancing Real World Online Discussions

After a while we decided to publish an overview paper containing and explaining the idea behind DBOA so far and how the developed tools are interacting with each other in Schneider and Meter 2019:

Alexander Schneider and Christian Meter.

"Various Efforts of Enhancing Real World Online Discussions"

In: *Reason to Dissent: Proceedings of the 3rd European Conference on Argumentation (ECA 2019), Volume III, ISBN: 978-1-84890-333-3, College Publications.*
Acceptance Rate: ~75%

Summary

In this paper we summarize our research findings and demonstrate that they are playing together nicely. At first, we give a comprehensive overview about DBOA and the ideas behind this approach, the first experiments and the field experiments. Technical details about the extensibility of our argumentation backend D-BAS are also described. Furthermore, we explain our view on "Arguments as a Resource" (see Chapter 5) and briefly describe a versioning system for arguments. A short description of our distributed network EDEN to exchange arguments and their relations follows.

Our efforts in integrating our systems into other web contexts and how they are using our previously described software tools is then put into context. At last, integration into Social Networks and conducting a dialog with users from Facebook is described. Exporting and importing data from other commonly used tools in the argumentation theory is presented with *dabasco* (Neugebauer 2018) as well.

Personal Contribution

This overview paper gives an overview about our research in the field of DBOA. Therefore, no new ideas are included and only previous development and research is presented.

The author of this thesis, Christian Meter, authored Section 2 ("Dialog-Based Argumentation"), Section 4 ("discuss"), Section 5 ("Experiences with Auxiliary Approaches") and Section 6 ("Related Work"). The remaining parts were authored by Alexander Schneider.

Importance and Impact on this Thesis

Having an overview paper describing our thoughts, software tools and how the projects can be combined, is necessary, in order to enable others researchers to understand the current state of our tech stack and of our latest findings. It also helped us to define how far we reached with our findings on the way to rethink the way people are discussing on the Internet.

Various Efforts of Enhancing Real World Online Discussions

ALEXANDER SCHNEIDER
Heinrich-Heine-University Düsseldorf, Germany
alexander.schneider@hhu.de

CHRISTIAN METER
Heinrich-Heine-University Düsseldorf, Germany
christian.meter@hhu.de

In this work we present a suite of software which enables gathering of natural language arguments from non-expert users of argumentation software without the use of NLP or other argument mining techniques. This is achieved by presenting the user with interfaces that prompt them to enter the data in a way in which it can be correctly added to an argument graph.

KEYWORDS: argumentation, online discussions, dialog-based discussions, web applications

1. INTRODUCTION

In this work we present various efforts that try to answer the question of how to gather structured argumentation graphs from natural language discussions of non-expert users.

Gathering arguments through argument mining from natural language is an ongoing research effort that made a lot of progress in the last years. Despite this, considerable challenges need to be solved before argument mining is at its peak. Because of this we present different ways of gathering argument data from natural language discussions.

We tackle the problem by designing interfaces and systems which allow the user to input arguments, while the data is automatically structured into an argument graph in the background. We made several efforts to design dialog-systems which make use of this approach to interact with everyday users that are not argumentation-experts in any way.

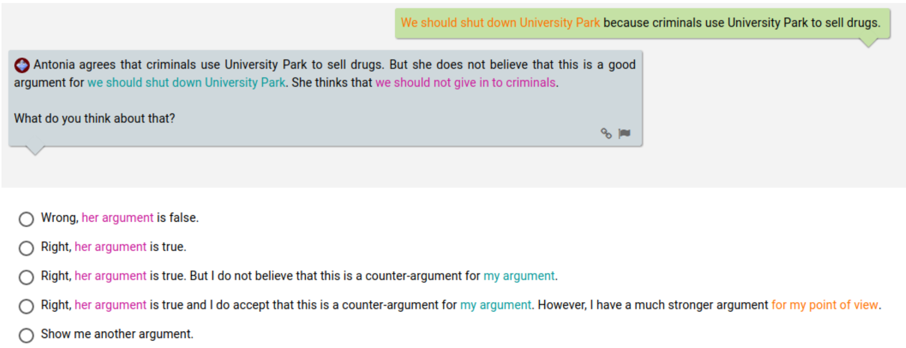


Figure 1: Gathering feedback during a confrontation in D-BAS.

A typical user is presented with an argument and the request to react to that argument (see Figure 1). Participating users can then position themselves to that argument using statements introduced by other participants, thus strengthening the existing graph-structure or enter their own opinion. In that case the interface prompts them to input their argument in such a way that structured argument data is produced without further processing. This can be done by adding the new statement in the proper place in the argumentation graph, which the system can deduct from the selected choices of the user.

In this paper we describe three such interfaces, namely our *Dialog-Based Argumentation System* (D-BAS) (Krauthoff et al., 2018), *discuss* (Meter et al., 2017), and *Jebediah* (Meter et al., 2018). The interfaces differ in their approaches and use-cases. While D-BAS is a dedicated webservice for discussions which the user needs to visit, discuss allows the embedding of the interface into arbitrary websites. Jebediah enhances user experience by providing an agent for social networks with support for natural language processing. All these approaches share the same argumentation engine in their backend, which is accessible via D-BAS' *Application Programming Interface* (API) in the reference implementations.

The structured data created by the interfaces lends itself to reuse, and as a consequence we also present our *Extensible Discussion Entity Network* (EDEN) (Meter et al., 2018). EDEN is a reference implementation, which be used by discussion-providers to perform an automatic exchange of argumentation data. Examples of exchanged data are statements and arguments from the users, which can then be re-used in further discussions. We show that (automatic) reuse of argument data is possible and valuable.

As a last step in our pipeline we also provide a tool called *dabasco* (Neugebauer, 2018), which enables the transformation of the gathered data into instances of *Argumentation Frameworks* (AF) (Dung, 1995), *Abstract Dialectical Framework* (ADF) (Brewka & Woltran, 2010) and ASPIC+ (Caminada & Amgoud, 2007).

Thus, we present a complete pipeline of software projects which aid in the creation of natural language online discussions for non-expert internet-users, resulting in structured argumentation graphs that can be further used for analysis and other relevant processes. We reason that the pipeline presented in this paper is viable in conducting large-scale online discussions.

The rest of the paper is structured as follows: Section 2 describes dialog-based argumentation in general and D-BAS in particular. Following, Section 3 introduces the reuse of arguments and an implementation for networking several dialog-based argumentation systems. In Section 4 an alternative interface for integration of dialog-based argumentation systems into arbitrary web content is discussed. A social-agent-based interface and miscellaneous ways of exporting the collected data into other discussion frameworks are presented in Section 5. In closing, we discuss related work in Section 6 and end with our conclusions and future work in Section 7.

2. DIALOG-BASED ONLINE ARGUMENTATION

A lot of research in the argumentation community focuses on argument mining from natural language texts. Most argument mining research is done with the goal of creating a machine understandable corpus of arguments, which can be processed and used by algorithms. With that same goal in mind, we want to present a different approach. Instead of letting human users debate with free text, e.g. in forums, and trying to mine the arguments after the fact, we want to engage them in a dialog-like exchange. This exchange still lets the users use natural language but presents them with certain prompts at the same time. This compels the user to enter their thoughts in a structured manner, yielding arguments which can be added to an argumentation graph instantly.

2.1 The Idea Behind Dialog-Based Online Argumentation

Dialog-based argumentation was introduced in detail by Krauthoff et al. (2016, 2018) and is best described as a multi-user dialog with a single system. Each user is confronted with an argument for some topic, that was not generated by the system but was entered by other users. Therefore, the user is basically engaged in a time-shifted dialog with other users. The main difference to "traditional" online discussions like forums is that the user is at all times being presented with a single argument, instead of e.g. a list. After the user reacts to the presented argument, a next argument made by other participants is chosen based on the user's reaction. The reaction is then stored to be used in future interactions with the system.

Let us take a look at an example: The system contains a discussion with the topic “We should renovate the city’s library”. Now, the system could present the interested user with several options, which confront the user with arguments *in favor of* renovating the city’s library or with arguments *against* renovating the library because, for example, it costs too much money. The user in turn can react to those arguments by either choosing counter- and supporting arguments that other users already made, and the user feels are compelling, or by entering their own thoughts. This step is the crucial one which prompts the user to enter their argument in a structured manner as presented in Figure 4. Since the user is guided through a specially crafted menu, the system knows whether to input the user’s statement as an attack or support on a certain other statement, or if it is e.g. an undercut for some argument.

2.2 User-Focused Measures

The type of argument gathering, that we present with dialog-based discussion, relies heavily on the correct use of the system by the users. This leads us to focus on interface measures, which help the participants to navigate the system without issues.

Let us say a user is interested in the topic of whether to buy a dog or a cat. After the user expresses their interest in the topic, the system asks the user about what they want to debate in detail. Those options are for example “We should get a dog”, “We should get a cat” or “We should get another pet”. When the user selects the position they are interested in, they are prompted to state whether they are in favor or opposed to that option (or have no opinion but want to see some arguments for that option). This is done, so the system knows whether the user interactions to come should be tallied as attacks or supports of certain arguments. Furthermore, it enables the system to confront the user with fitting arguments from its database.

Anytime the user formulates their own arguments instead of reusing others, the system scans for similar arguments already made and presents them to the user. They can then choose to use one of the already present arguments to keep duplicates to a mini- mum. The dialog continues until the user does not want to have a discussion anymore, or until they reach a point in the discussion graph where there is no more attacking or supporting arguments left.

Duplicate, malicious or grammatically unsound arguments still make it into the system, since its main input source are typical humans.

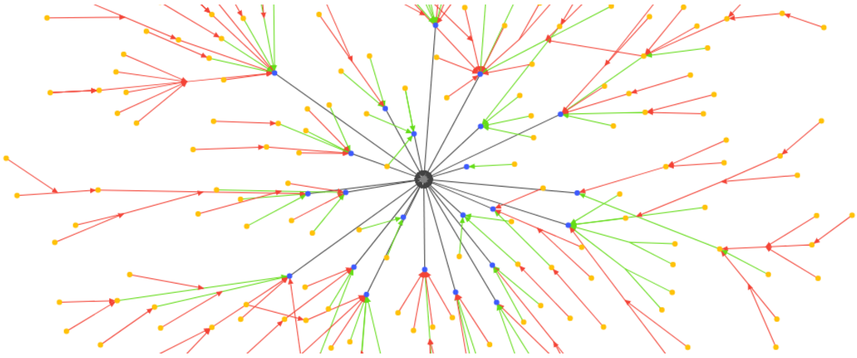


Figure 2: The graph resulting from discussions through D-BAS, discuss and Jebediah. Depicted is an instance from a real-world discussion. Colors: grey: issue, blue: positions, yellow: statements, green: supports, red: attacks.

Those arguments can be moderated to make the experience a pleasant and engaging one for the users. Instead of using traditional moderators, the system implements the power of the masses. This has been included in D-BAS as a *decentralized moderation system* (Krauthoff et al., 2018). Users can e.g. mark duplicates or arguments violating the community's policies. Experienced users can then visit special randomized moderation queues, where they are presented with some of the marked arguments and can democratically vote whether to take action against those. Possible actions are for example "delete argument", "reformat argument" or "merge duplicates". If enough votes are tallied for a single option, it is executed.

2.3 Field Experiences

The dialog-based argumentation system D-BAS is online and free to use[1]. Besides experiences gathered from running the service, there also have been lessons learned from a formal evaluation through a field-study (Krauthoff et al., 2017). The study took place over 19 days and had 318 unique participants that visited the corresponding website. In this study the topic was how the computer science faculty could improve the bachelor's courses despite student numbers growing rapidly. All computer science students were invited to participate, and the faculty promised to use the results as a base for future decisions.

During the experiment, more than 250 arguments have been created, which seems to suggest that users untrained in argumentation techniques are able to create a complex argument graph with the help of

[1] https://dbas.cs.hhu.de

dialog-based argumentation. Parts of the resulting graph can be seen in Figure 2 and the associated data can be obtained online.

2.4 Application Programming Interfaces

D-BAS has two fully documented[2] and usable API options built-in to export the contents of a discussion and to allow third party applications to access the *Dialogue Game Execution Platform* (DGEP) (Bex et al., 2014) parts.

The first endpoint provides authentication, authorization and the execution of discrete steps in the discussion. Applications can send requests to this endpoint to tell D-BAS about their current status of the discussion which then produces a response containing the next options and possible next discussion actions. Also sample text-responses are returned, which can then be used.

Data retrieval from our databases can be achieved using the second endpoint, which provides a GraphQL (The GraphQL Foundation, 2019) API. This way people interested in the data can write their own queries to our databases to retrieve the public information from the hosted discussions.

3. NETWORKED ARGUMENTS AS A RESOURCE

Through the use of dialog-based argumentation, people are able to create a wealth of arguments by following a dialog. But there are also scenarios where D-BAS has disadvantages. If we assume that, for example, several media outlets use dialog-based argumentation instead of simple list-like comments under their publications, each of them could run their own instances of dialog-based argumentation software. Now, every user that wants to debate the same or a similar topic at different media outlets, is confronted with repeating arguments they are already familiar with. This would almost certainly happen due to the nature of how dialog-based argumentation is conducted. Furthermore, arguments made at one instance will never be seen on another, no matter how insightful or well worked out they may be. This section presents our thoughts on how to tackle these and related challenges.

3.1 Distributing and Versioning Arguments

We call every host, from the before-mentioned scenario, running their own dialog-based argumentation software, an *aggregator*. To put it in another way: an aggregator is an entity providing content and the space

[2] https://dbas.cs.hhu.de/docs

to discuss it. To allow distribution of arguments, every aggregator can join a distribution network. Aggregators may have differing policies about which arguments are valid according to some rules or community standards. Hence, flooding the arguments to all aggregators in the network is unwise, because not all instances have the same policies and would be willing to receive certain arguments. Moreover, aggregators possibly want to keep the intellectual rights on arguments devised on their platform. Thus, every argument needs to reference which aggregator is the authoritative instance for it. This means, that the arguments stay property of the differing aggregators, but still can comprise a single argumentation graph spanning over different physical and logical entities participating in the argument network. To allow other participants to propose changes to arguments, that they are not authoritative of, we need to introduce versioning. As presented by Meter, Schneider and Mauve (2018) one can use a decentralized version-tree which is already known for versioning source-code. This means, that every argument has a pointer to its predecessor if one exists. Any changes can be proposed at once without violating or changing the original argument by creating a changed version which points to the original as its predecessor. The authoritative aggregator can decide whether to accept any of the proposed updates and incorporate them into the official version. But even in that case, there will be a new version from the authoritative source, since all arguments are created immutable.

3.2 EDEN: Extensible Discussion Entity Network

An exemplary implementation of a distributed argumentation network powered by aggregators is EDEN which was presented in detail by Meter et al. (2018). EDEN was developed in Clojure, a functional language on the JVM. Furthermore, we pursued a modular approach with EDENs architecture, which splits it up into four distinct modules – interface, discussion platform, database and aggregator core – which can be interchanged as long as the new module adheres to the proposed interfaces between the major parts.

The interface is tasked with guiding the user through the dialog-based argumentation. A database stores and persists the locally needed arguments. It can also provide features like semantic search on the arguments. The discussion platform is the piece of software that provides the internal logic on how to conduct the dialog-based argumentation, also known as DGEP. In the default case EDEN utilizes D-BAS as a DGEP. An aggregator core coordinates the flow of arguments between the different modules as well as between aggregators.

Communication between aggregators is handled in two parts. First, there is a REST API providing aggregators with the ability to actively query for discussion entities like arguments and their interrelations. As a second option a publish/subscribe queue exists, which automatically updates entities from known aggregators. For example, if aggregator *B* requests some argument *X* on the topic of dogs from aggregator *A*, they also subscribe to the corresponding queues. When an update for *X* is available, *B* automatically gets informed about the update by *A* via the queue. Different update forms can be used. Instead of updates on queried arguments, *B* could receive notifications every time there is a new argument on the topic of dogs, to broaden its repertoire

4. DISCUSS: EMBEDDING DIALOG-BASED ARGUMENTATION INTO WEB-CONTEXTS

One of the first applications using the API of D-BAS, is *discuss* (Meter et al., 2017). discuss provides a minimal discussion interface to interact in the same flow as we have seen it in D-BAS, with the distinction, that it can be embedded in every web-context utilizing a JavaScript environment. This is intended to be used, for example, in online newspaper articles, which ask the readers to start a discussion in the comment sections. But since comment sections do not provide any structure, this approach could bring a significant improvement, because of the structural manner how the arguments of the users are being gathered.

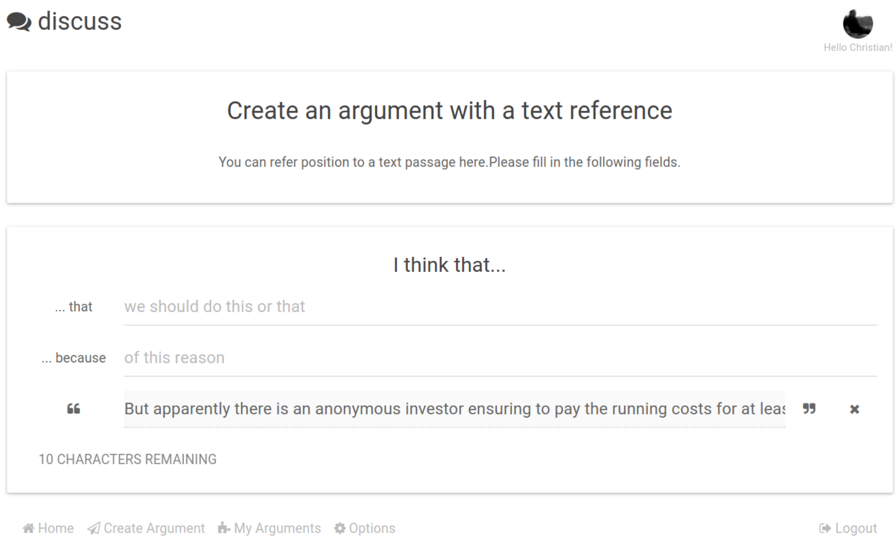


Figure 3: discuss: Create a new argument with a reference to a passage in the author's article.

Without having to leave the current scope, discuss provides (1) direct interaction with the author's arguments, (2) jumping into the discussions, where other participants interacted with the article, (3) enabling discussions in our proposed dialog-based flow (see Subsection 2.1) and (4) connect to the EDEN network.

4.1 Interacting with the Author's Arguments

One of the core functions of discuss is to directly interact with the author's article. Selecting an interesting part of a text passage opens up a dialog, where the reader can create a new argument with the selected text as a *reference* (see Figure 3). Internally, the creation of an argument in this way is the same procedure as adding a new position in D-BAS, which introduces a sub discussion in the context of the discussion topic.

4.2 Jumping into the Discussion

Interactions with the article, which created a new argument with a reference to parts of the article, are highlighted so that the user sees an interactive element on the website (see Figure 4). These references provide an entrypoint to the discussion, where the user's argument has been used. Also, other arguments, which referenced the same text passages, are listed and users can decide where they want to jump into the discussion.

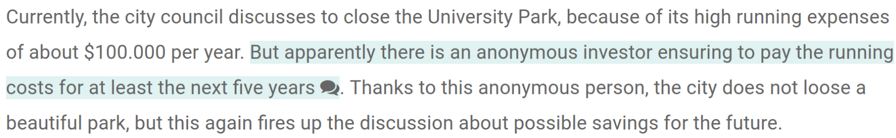

Currently, the city council discusses to close the University Park, because of its high running expenses of about $100.000 per year. But apparently there is an anonymous investor ensuring to pay the running costs for at least the next five years. Thanks to this anonymous person, the city does not loose a beautiful park, but this again fires up the discussion about possible savings for the future.

Figure 4: Text passage from an article, which has been used in an argument. A click on it opens the interface to jump into the discussion

4.3 Dialog-Based Discussion Flow

We omit the selection of the initial positions in discuss, because we encourage to directly jump into the discussion via a reference in the text, i.e. hook into a pre-existing argument from a user, or by selecting a text-passage, i.e. create a new argument referring to the text. After the initial step, discuss presents the classical discussion flow which we have already seen in D-BAS (see Subsection 2.1). Specifically, this means that we conduct a dialog with the users and present those arguments, which have been posted about the argument from the article.

4.4 EDEN Integration

Besides the described functions, discuss can be used to connect to the EDEN network (see Subsection 3.2). D-BAS is then solely used as an DGEP for the steps in the discussions, whereas the arguments are being fetched from EDEN. This mechanism allows to retrieve and collect arguments from different locations and discussions, which can then be used in the current article's discussion.

5. EXPERIENCES WITH AUXILIARY APPROACHES

Based on the presented tools, we felt the need for auxiliary applications. One is *Jebediah*, an alternative interface into dialog-based online discussions enabling users to discuss matters through chatbots and voice assistants. Furthermore, we present *dabasco*, which allows the data generated through D-BAS and its applications to be converted to other discussion frameworks for further use.

5.1 Jebediah

A vast part of online discussions takes place on social media platforms. Jebediah (Meter et al., 2018) is an interface which enables users of those platforms to take part in dialog-based online argumentation through chat-bots and voice assistants. Classifying the user's input is realized with the help of Google's Dialogflow platform (Google Ireland Limited, 2019), which is an Artificial Intelligence processor that tries to match the natural language input against predefined and pre-trained rules. The matching-process has the goal to produce structured data and the resulting data is being sent to a dialog-based argumentation software, like D-BAS. It returns a response, which is then again formatted and forwarded to the user through the chat-bot (see Figure 5). This is still a highly experimental feature, which works most of the time but certainly can be further improved upon. Nonetheless, it would be interesting future work to test how users feel when discussing topics with a bot instead of a text-interface.

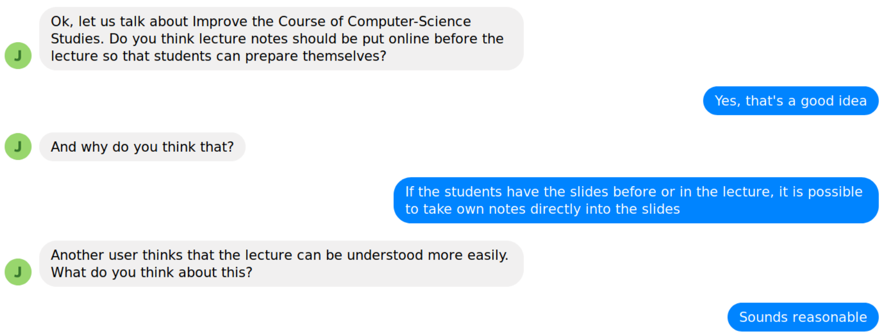


Figure 5: Left side: Dynamically produced text messages from Jebediah, right side the user's answers in the Facebook Messenger.

5.2 dabasco

The last step in our pipeline is the export of the generated data. Exports are useful to utilize collected argument data for further analysis. Building on the fact that some established tools by the community expect certain formats, Neugebauer developed an export interface called *dabasco* (Neugebauer, 2018). This way it is possible to export AF, ADF, and ASPIC+ data which was converted from D-BAS' data structure. dabasco uses D-BAS' API and provides the first 3rd party application interacting with our software stack.

6. RELATED WORK

Tools for facilitating online argumentation have been described and developed before. The set of tools that is most like the proposed pipeline is the argument web (Rahwan et al., 2007). We build on similar ideas of a unified structured web of arguments and are not striving to compete with the argument web but to be compatible to magnify the extend of the argument network. AIFdb, developed by Lawrence et al.

(2012), is in spirit akin to EDEN regarding collecting arguments from differing sources, but differs in aspects of centralization and the kind of arguments collected. Other approaches at structuring arguments, include Carneades (Gordon & Walton, 2006), Deliberatorium (Klein & Iandoli, 2008) or OVA as introduced by Snaith et al. (2010). The difference to is that none of those are based on dialog-like argumentation. Most of these tools focus on the whole discussion, whereas our smallest entity is the statement, which could be put together to an argument and the put into context, e.g. of a discussion.

7. CONCLUSION

In this paper we presented a complete pipeline for gathering, sharing and exporting user- generated arguments. We introduced D-BAS, a system that conducts discussions by simulating a dialog with other users. A field-study verified that this approach yields a structured argumentation graph and even untrained users were able to use our software in a productive way. Moreover, we presented discuss, which enables arbitrary websites to integrate a D-BAS-style discussion and Jebediah, which does the same for artificial assistants. To share the generated arguments between instances of D-BAS, we use EDEN, which provides the ability to decentralize an argumentation network. Lastly, dabasco allows the export of D-BAS arguments to different argumentation frameworks, which can be used for further calculations.

This paper showed that a pipeline for gathering structured argumentation from natural language without argument mining is possible and how such a pipeline may be structured.

For future work we plan to conduct field experiments that make use of the complete pipeline to test its efficiency. We furthermore are developing tools that harness the dialog-based stack to conduct discussions with the goal of finding and voting on solutions for e.g. the budgetary allocation of a city.

REFERENCES

Bex, F., Lawrence, J., & Reed, C. (2014, January). Generalising argument dialogue with the Dialogue Game Execution Platform. Frontiers in Artificial Intelligence and Applications 266, pp. 141-152.

Brewka, G., & Woltran, S. (2010). Abstract Dialectical Frameworks. Proceedings of the Twelfth International Conference on the Principles of Knowledge Representation and Reasoning.

Caminada, M., & Amgoud, L. (2007). On the evaluation of argumentation formalisms. Artificial Intelligence, vol. 171, pp. 286-310.

Dung, P. (1995). 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.

Google Ireland Limited. (2019, June). Dialogflow. Retrieved from https://dialogflow.com/

Gordon, T., & Walton, D. (2006). The Carneades Argumentation Framework - Using Presumptions and Exceptions to Model Critical Questions. Frontiers in Artificial Intelligence and Applications, vol. 144: Computational Models of Argument, pp. 195-207.

Klein, M., & Iandoli, L. (2008, February 20). Supporting Collaborative Deliberation Using a Large-Scale Argumentation System: The MIT Collaboratorium. MIT Sloan Research Paper No. 4691-08.

Krauthoff, T., Baurmann, M., Betz, G., & Mauve, M. (2016). Dialog-Based Online Argumentation. Computational Models of Argument, pp. 33-40.

Krauthoff, T., Meter, C., & Mauve, M. (2017). Dialog-Based Online Argumentation: Findings from a Field Experiment. Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence, vol. 2012, CEUR-WS, pp. 85-99.

Krauthoff, T., Meter, C., Betz, G., Baurmann, M., & Mauve, M. (2018, September). D-BAS – A Dialog-Based Online Argumentation System. Computational Models of Argument, pp. 325-336.

Lawrence, J., Bex, F., Reed, C., & Snaith, M. (2012). AIFdb: Infrastructure for the Argument Web. Frontiers in Artificial Intelligence and Applications, vol. 144: Computational Models of Argument, pp. 515-516.

Meter, C., Ebbinghaus, B., & Mauve, M. (2018, September). Jebediah – Arguing With a Social Bot. Computational Models of Argument, pp. 467-468.

Meter, C., Krauthoff, T., & Mauve, M. (2017, July). discuss: Embedding dialog-based Discussions into Websites. Human Computer Interaction International: Proceedings of HCI, pp. 449-460.

Meter, C., Schneider, A., & Mauve, M. (2018, September). EDEN: Extensible Discussion Entity Network. Computational Models of Argument, pp. 257-268.

Neugebauer, D. (2018, September). DABASCO: Generating AF, ADF, and ASPIC instances from Real-World Discussions. Proceedings of the 2018 Conference on Computational Models of Argument, pp. 469-470.

Rahwan, I., Zablith, F., & Reed, C. (2007). Laying the foundations for a World Wide Argument Web. Artificial Intelligence, vol. 171, no. 10-15, pp. 897-921.

Snaith, M., Devereux, J., Lawrence, J., & Reed, C. (2010, August). Pipelining Argumentation Technologies. Proceedings of the 2010 conference on Computational Models of Argument, pp. 447-453.

The GraphQL Foundation. (2019, June). GraphQL - A query language for your API. Retrieved from https://graphql.org/

3.4 Further Findings: Dialog-Based Discussions to Support Decision-Making Processes

D-BAS was used in another field experiment at Heinrich-Heine-University, that is described in the master's thesis by Ebbinghaus 2019, which was supervised by the author of this dissertation, and will be published in a demo session in Ebbinghaus and Mauve 2020.

The goal of this experiment was to have computer-science students discuss and decide about € 20,000 and how those could be used to improve the computer-science studies. Proposals were collected as positions and the students had two weeks time to add and discuss these proposals. After this phase, we condensed the proposals and merged them, if they were semantically identical or comparable. One more week was given to the students to discuss the reworked proposals. Afterwards, the students had to use *decide*[1], a web application to vote for their favorite proposal connected to D-BAS via API.

Overall, the students were able to use DBOA in order to form an opinion about the possible proposals and to finally cast a vote. The DMS was intensively used to keep a thread running through the debate of the users. Also, we could reduce redundant arguments by presenting users previously discussed arguments, which were then reused in the current discussion.

In the end, the students decided for which projects the money should be spent, e.g. creating a free hacker space or providing power strips for lecture halls.

This project underlines once more the practical usability of dialog-based discussions. See further findings and more details in Ebbinghaus 2019 and Ebbinghaus and Mauve 2020.

[1] http://github.com/hhucn/decide

Chapter 4

Integrating Dialog-Based Online Argumentation

After our dialog-based approach has been implemented and field-tested, we use this technique in other web-contexts to leave the scope of a standalone web-application. This aims at bringing dialog-based discussions to more locations on the Internet. Therefore, we extended D-BAS to provide an API, which can be used by more tools to enable dialog-based discussion in Social Networks or directly in online news media articles.

One major challenge is to re-think bootstrapping a discussion. It needs to be more lightweight and easier for the participants to confront other users. In the "original" dialog-based approach (described in Chapter 3), users started from the very beginning of a discussion, having to choose a position to finally be able to participate. Jumping to the right position or joining the conversation based on personal relevant keywords is necessary to enable productive and fast discussions.

We propose two different styles to enable side-entries into a discussion in the next sections: either via text-reference (Section 4.1) or via natural language processing (Section 4.2). Finally, Section 4.3 describes a lab-experiment with participants, who had to discuss either with our tools or with conventional comment-section tools.

4.1 discuss: Embedding Dialog-Based Discussions into Websites

We are now concentrating on the contributions and the impact of the following paper (Meter et al. 2017a):

Christian Meter, Tobias Krauthoff and Martin Mauve.

"discuss: Embedding Dialog-Based Discussions into Websites"

In: *Proceedings of the 4th International Conference on Learning and Collaboration Technologies, held as part of HCI International 2017*, Volume 10296 of *Lecture Notes in Computer Science*, pages 449–460, Springer.
Acceptance Rate: ~28%

Summary

The first application to use D-BAS as an argumentation backend is discuss. It is a web application that can be embedded into an arbitrary context, which allows the execution of JavaScript code. At first, we focus on the integration into online articles, e.g. articles of newspapers, to provide a possible comment section replacement. The goal is to include DBOA into the article to enhance the overall discussion experience and to reduce the common problems of current online discussions as they are described in Section 2.1. We achieve this by conceiving and developing the lightweight interface discuss and integrating it into the website. discuss does not provide an implementation of a DGEP, but uses D-BAS' API instead to enforce dialog-based conversations. Therewith, users can now conduct structured discussions without having to leave the website they are currently visiting.

A new feature is to create text references and to integrate these selections into ones own argumentation. With this, a user's argument can be supported by, for example, a fact from the article itself. Deep and direct interactions with the article are possible now.

The text references are thus a simple entry point into the discussion. Sidestepping into a discussion is not trivial, because our system does not know what the new user is thinking about and at what point the user wants to join the discussion. Therefore, we implemented a *jump-interface*, which asks for the necessary information to guide the user to the correct position in our argumentation graph (see Figure 9 in Meter et al. 2017a).

At last, the paper closes with conclusion, outlook and information about the technical foundation which enables the embedding of the software into common web environments.

Personal Contribution

The initial idea and a complete implementation of a condensed interface for a dialog-based system discuss stem from Christian Meter. Sections 2, 4, 5, 6 and 7 were authored by Christian.

Tobias Schröder (né Krauthoff) wrote the introduction (Section 1) and explanations of DBOA (Section 3). Furthermore, the ideas of the jump-interface were developed by Christian, whereby Tobias provided further considerations about the interface and implemented the necessary changes in D-BAS. Martin Mauve reworked the introduction, provided his thoughts on the integration of DBOA and gave feedback on the paper.

Importance and Impact on this Thesis

With this paper we present the first real-world application encouraging dialog-based discussions in a global medium. We leave the controlled scope of a standalone website and enable arbitrary usages of this software. The condensed interface, which is as expressive as the original version in D-BAS, allows to use the complete feature-set provided by DBOA in browsers. It also shows that tight interaction with the articles themselves is necessary for more direct discussions about its content.

We still need to verify how this interaction with the article and the whole idea about dialog-based discussions in different web contexts can be used by untrained users. We are working on this and will present results of a separate study in Section 4.3.

discuss: Embedding Dialog-Based Discussions into Websites

Christian Meter, Tobias Krauthoff, and Martin Mauve

Department of Computer Science
Heinrich-Heine-University Düsseldorf, Germany
{meter,krauthoff,mauve}@cs.uni-duesseldorf.de
http://cn.hhu.de

Abstract. In this paper we present the web application *discuss*, which provides a novel approach to embed structured discussions into any website. These discussions employ a formal argumentation system in their backend and can be used in addition to or replace existing comment sections. By interacting with the content of the website, we allow to include this content in the discussion. Furthermore, the same discussions can be accessed from multiple websites to bring their audiences together and create a single large discussion. To form a combined audience, it is necessary to use a common backend and we present an exemplary implementation of this scenario.

Keywords: online argumentation, collaborative work, discussions, dialog-based approach, web technologies, computer science

1 Introduction

Many websites and online news media provide their readers with the opportunity to comment and discuss their content. In fact, the ability to participate in such a discussion or to read what others think about an article is a major reason to prefer online content over offline media. While current solutions are quite suitable to provide simple feedback, they do a rather poor job at fostering meaningful discussions among the readers. This is especially true in those cases where this would be most needed: for articles that receive a lot of reader-feedback due to their popularity or controversial nature.

Commonly, comment sections are located below online articles. They provide a vertical-oriented discussion, where one comment follows the other, often combined with the possibility to directly reply to an individual comment. This is the same design used, for example, by Facebook or Twitter or, in fact, in most forum systems. It is well known, that this design has significant flaws when used for discussions and argumentation rather than simple feedback [1,2], for example redundant comments, lack of structure or simply missing scalability when large numbers of users try to express their opinions. Some online editorials, e.g. The Guardian, are really interested in the comments from the users to enrich the journalism, but often they are abusive, violate their community standards and the journalists are confronted with huge numbers of comments, which they have to moderate [3]. In general, online editorials show keen interest in the discussions in the comments and are interested in the user's opinions.

To solve these problems and allow for meaningful online argumentation regarding issues raised in an online news media article, we propose to integrate *dialog-based online argumentation* in the website hosting the article. In dialog-based online argumentation the user performs a time-shifted dialog with those users who previously participated in the discussion. The new user can then react to statements from those other users and provide her own statements. This dialog is performed in natural language and the user does not need any specific skill other than being able to read and write. This concept has been implemented in the argumentation system *D-BAS* [4], which is a public accessible web-application. The system also provides an application programming interface to use its backend to remotely perform steps in the argumentation.

In this paper, we present *discuss*, which uses the interface of D-BAS to embed structured discussions in arbitrary websites. discuss is a JavaScript-based extension, which can seamless integrate dialog-based discussions into websites. This tool can be used to enhance or replace existing comment sections whenever a discussion is intended to be held with or among the readers. It gives users that participate in the discussion the option to add references to parts of the online article to their statements. Those parts are then marked in the article, so that other readers can jump right into the ongoing discussion. Furthermore, it is possible to browse and search for those arguments in the discussion that reference the current website.

Our main contributions in this paper are: (1) integrating the interface for dialog-based online argumentation into regular web-content, (2) allowing for references between the argumentation and the content of the website, (3) navigating the argumentation by means of links and search requests and (4) providing a way to use the same discussion across multiple websites.

The paper is structured as follows: Section 2 contains the related work to compare our approach to existing established technologies in the Web. Section 3 is about the prototype D-BAS and the concept of dialog-based argumentation. Section 4 describes the functionality of discuss, while Section 5 focuses on our implementation. The last Sections 6 and 7 conclude the paper and give an outlook to future work.

2 Related Work

The most popular tools to provide reader feedback are simple comment sections in form of a linear list of user statements or the use of forum-based systems. Both display all the negative aspects mentioned above. There are three specific systems that we want to discuss in more detail:

The first system is *Disqus*, which enables discussions on arbitrary websites [5]. In fact, Disqus is a JavaScript application, which needs to be installed by webmasters and brings a hosted alternative to self-hosted comment sections. One unique characteristic is that instances from different websites can discuss about a global topic. Disqus does not introduce new techniques to enhance discussions and, in general, provides the same functionality as normal comment sections, i.e. add, reply to and vote on comments. This tool is popular for its simplicity and is therefore used quite frequently. It does not address the common problems of comment sections, though. Enabling a global discussion, however, is quite interesting and will also be used in our application.

rbutr [6] is a browser extension which gives the users the ability to link several websites sharing a common topic. These links can then be combined with arguments to introduce information from website *B*, which might support or rebut the article presented on website *A*. When a user then visits website A, she is presented a small popup showing that B provides arguments against the contents of A. Therefore, rbutr can be used to link contents from different websites to adjust false information presented on another website. The general idea of using contents from the Internet to support one's own statement is also used in discuss.

ArguBlogging from ARG-tech [7] can be installed as a bookmarklet[1], which needs no further configuration and can directly be used by interested users. The main concept of this tool is to select arbitrary text passages from websites and post them with a reference to the original source on one of the supported blogging sites, currently *tumblr* and *Blogger*. ArguBlogging then creates a post on the user's personal blog and gives her the ability to discuss about this text passage. A popup is presented to other users, who use ArguBlogging, when they arrive on a website, where another user already has selected some text and discussed it on her blog. These other users can then react to this statement and join the discussion. The idea behind the text-selection feature from ArguBlogging is also used in discuss, but in our case it will be directly integrated into a dialog-based discussion.

3 Dialog-Based Online Argumention

The goal of dialog-based online argumentation is to enable any user to participate efficiently in a large-scale online argumentation. At the same time it avoids, or at the very least reduces, the problems that occur in unstructured online argumentation such as a high level of redundancy, balkanization, and logical fallacies.

In the following, we briefly describe terms that will be used to explain the main aspects of dialog-based online argumentation. Based on these terms, we then introduce the main concepts of dialog-based online argumentation.

Each discussion is a set of *statements*, which are the most basic primitives used in an online discussion. The negation of a statement is itself a statement. Individual participants might consider a given statement to be true or false. A *position* is a prescriptive statement, i.e., a statement which recommends or

[1] `http://www.bookmarklets.com/about/`

4

demands that a certain action can be taken. Further on we need to distinguish between first-order and second-order *arguments*. A first-order argument consists out of a premise group — a set of at least one statement — and a conclusion, i.e. a statement. Both are connected by an inference, which is either supporting or attacking, so that the premise group is a reason for or against the conclusion. A second-order argument has the same kind of premise group, but the conclusion is the inference of an argument. With this we can argue about the validity of another reason-relation. Together, the arguments of a debate form a (partially connected) *web of reasons*.

The core idea of dialog-based online argumentation is a loop consisting of three steps: (1) presenting a single argument; (2) gather feedback from the user based on a list of alternatives and (3) the system selecting the next argument that is shown to the user based on the response and, possibly, the data gathered from the responses of other participants [4]. In this way the user and the system perform a *dialog* where the system selects arguments that are likely to be of interest to the user and then the user provides feedback on those arguments.

A first thing that the system needs to do when a new user wants to participate in the online discussion is to choose an initial argument. This is challenging since the system has no information on the user, yet. One fairly straightforward solution is to simply ask the participant for an initial position she is interested in. After she has chosen or provided her position, she is asked to select or provide a statement explaining her choice. This statement is used as the premise, whereas the position forms the conclusion.

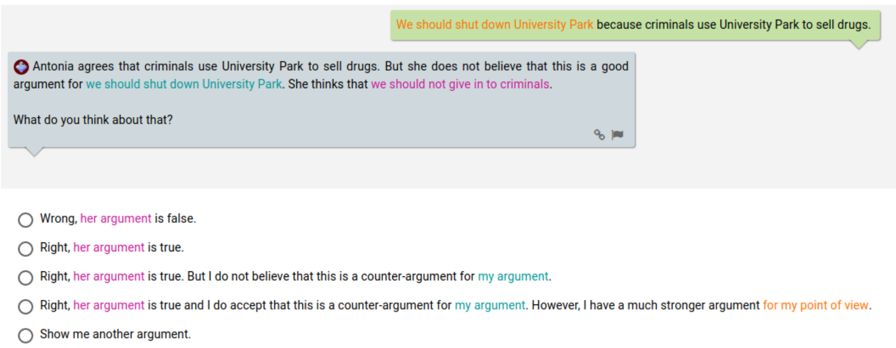


Fig. 1: Gathering feedback during a confrontation in D-BAS.

Once a user is confronted with an argument (see Fig. 1), she can provide feedback on the argument, as shown in Figures 1 and 2. Based on the feedback the system then selects the next argument that is shown to the user. A first prototype implementing this idea is the dialog-based argumentation system (D-BAS) which is available for testing[2].

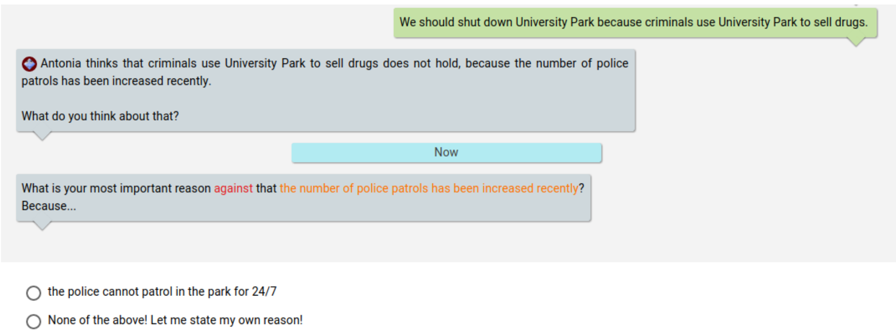


Fig. 2: Justification of the opinion in D-BAS.

4 Functionality of discuss

The idea of discuss is to embed dialog-based online argumentation into regular website content. To describe our implementation in more detail, we use an example where a city wants to reduce its spending and asks the citizen to propose some actions (positions) and to discuss them in detail. A user provided the position “We should shut down University Park” and other users started to discuss this position. This is the current state and we will show through this example how discuss works.

4.1 Embedding discuss into Online Articles

Imagine we have a discuss-powered website and have an article about the situation of the University Park. This article contains facts about the future of the University Park, which other users have proposed to close to cut spending of the city. As an example we assume that the article contains information about an investor, who is going to bear the costs of the park for the next years. We also assume, that our exemplary reader already has knowledge about the ongoing discussion and therefore knows some arguments in it. This is not absolutely necessary, but simplifies the explanation of our contribution.

The user starts reading this article. On her way through, she finds an interesting fact, which she wants to integrate in the discussion about closing University Park. To this end, she selects the appropriate text from the article, e.g. “*But apparently there is an anonymous investor ensuring to pay the running costs for*

[2] `https://dbas.cs.uni-duesseldorf.de`

6

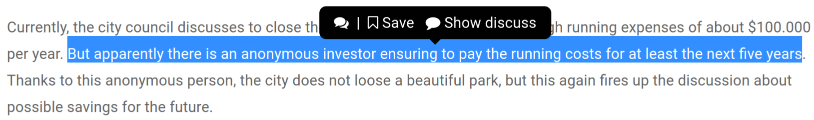


Fig. 3: Tooltip pops up when the user selects a text passage in the article.

at least the next five years". Selecting the text provides her with a *tooltip* (see Fig. 3). Possible options are "*Save*" and "*Show discuss*", where the first option stores the current selection in a *clipboard* for subsequent assembly of an argument for the discussion. The second option toggles the interface to discuss, so that she can directly participate in the discussion. To be flexible and not limited to specific websites, the interface is bound to a *sidebar*, which slides in from the right side, when the second option has been selected. In this sidebar all relevant elements are located which are necessary to participate in the discussion, see Fig. 4.

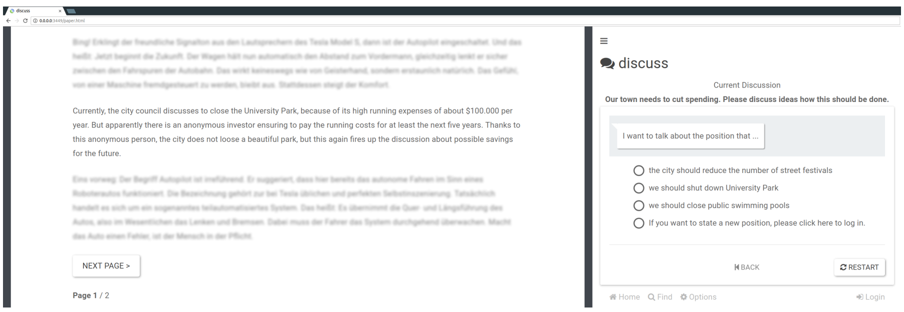


Fig. 4: Side-by-side integration of discuss into an online article.

Now, she can use the selected text and the interface of discuss to participate in the discussion and to create a direct citation of the text passage to her choice. We call these citations *text references* and the user can connect them with any statement in the discussion. With the knowledge the reader gained with this article, she is able to form a counter argument against closing the park and add a suitable reference to her statement. In this case her selection from above, pictured in Fig. 3, seems to be best-fitting, because it describes the future of the University Park in one sentence. These new facts are relevant and can stop the discussion about closing the University Park (if the sources of this article are trustworthy and the contents are true).

As a last step, the reader needs to add her argument to the correct location in the discussion. Since we are assuming, that she already has knowledge about the discussion, she can use the *search engine* for navigation. When the user now wants to add the fact that the investor is going to bear the costs, she needs

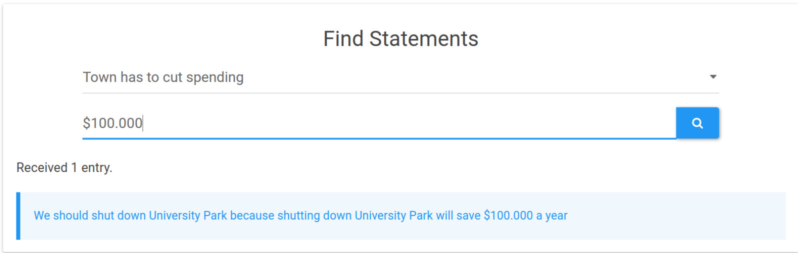


Fig. 5: Find position in the discussion, where the high costs of University Park is discussed.

to find the correct argument from the other user, e.g. "*We should shut down University Park, because shutting down University Park will save $100.000 a year*". Adding the exemplary input "$100.000" in the search engine (see Fig. 5) provides the statement we are looking for and we can now formulate our own argument against it supported by the reference from this article as it can be seen in Fig. 6. This completes the interaction with discuss and the user can close the sidebar to continue reading the article.

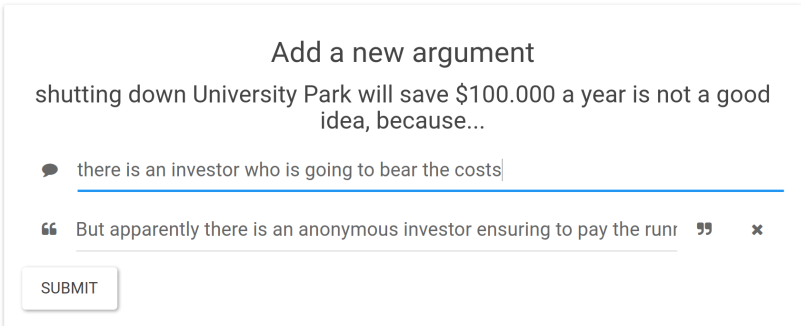


Fig. 6: Constructing a new argument with a text reference.

Arguments, references and their relations are stored in a common backend. All references from this article, which have been used in the discussion, are then highlighted in green color and appear in the text (see Fig. 7). Returning users or new readers of this article can easily see, that these text passages have been used in the discussion, and can interact with them by clicking on a reference. This click again toggles the sidebar and offers a simple interface with all linked locations in the discussion, where this reference has been used (see Fig. 8). Multiple locations are possible, since many users could use the same reference in their arguments.

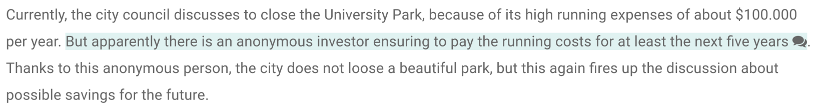


Fig. 7: Highlighted text reference which was previously used by a user.

Text references provide the easiest way to jump to a relevant position in the discussion and to directly start to discuss, because through a reference, our application presents the context of the related argument and asks the user how she wants to react to the argument, see Fig. 9.

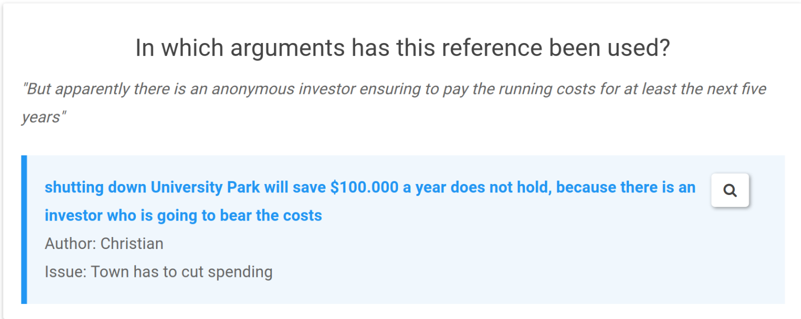


Fig. 8: Jump locations – shows where the references have been used.

4.2 Global Discussion

Common online news media websites, which provide a self-hosted comment section, only allow a local discussion. There is no possibility to leave the borders of this website to interact with users from other news media websites. Disqus [5] provides a feature for inter-website discussions, which we also included in discuss. To realize global discussions, we use one D-BAS instance as a common backend for websites that integrate discuss.

With these global discussions, a more heterogeneous peer group can be reached. Studies showed, that heterogeneous groups have a positive impact on the outcome of a discussion, i.e. solutions emerging from these discussions have a significantly higher quality and those solutions from homogeneous groups were never better compared to the heterogeneous group [8]. Therefore, enabling discussions among users from different online news media, with various levels of education and contrasting opinions, mutually support the discussion. Online news media are often known to have different audiences or specific political orientations and it could be very interesting to analyze discussions between those divergent peer groups, but this leaves the scope of this publication.

9

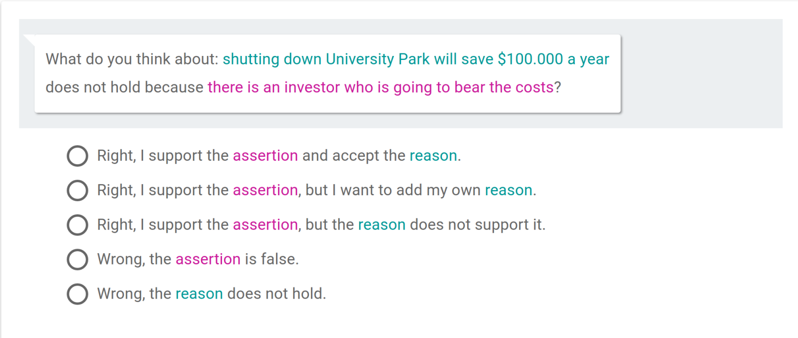


Fig. 9: Jump options – giving the user multiple options how she wants to react to the related argument.

5 Implementation

While implementing discuss we encountered a number of challenges that we outline in the following sections.

5.1 Technical Foundation

To create an application, which does not slow down existing websites and can pick any desired position in the *document object model* (DOM) of the website, we need to have powerful programming techniques and languages fitting our needs. The first prototype was implemented in pure JavaScript, but after few weeks the application became too complex and it was clear that we needed a framework to keep clean code and to reduce complexity. We were also unsatisfied with state-handling and the general language design of JavaScript, which is why we switched to the functional programming language *ClojureScript*[3] and re-implemented the functionality of the first prototype with just a few lines. ClojureScript compiles down to optimized JavaScript code with the *Google Closure Compiler*[4], which results in much faster code than we could manually develop. Using this compiler collection produces also much smaller production files thanks to advanced optimizations and dead code elimination. For dynamic user-interface handling, we chose Facebook's *React.js*[5].

These components allowed us to implement a stable and small web-application without disturbing or conflicting the website it has been embedded into. Since discuss adds many features and DOM manipulations as seen in the previous section, it is very important to choose the best-fitting components, because otherwise it would result in a slow or crowded application.

[3] `https://clojurescript.org/`
[4] `https://developers.google.com/closure/compiler/`
[5] `https://facebook.github.io/react/`

5.2 Including discuss in an Arbitrary Website

Website operators only need to include the to a single compiled and compressed JavaScript file to enable the features described in this paper for their websites. discuss searches in the DOM for a suitable entry point to enable dialog-based discussions. Selecting the text according to Subsection 4.1 is automatically available and the sidebar invisibly includes itself until the toggle in the tooltip is pressed. If an optional `div` is available in the DOM, an additional interface will be displayed on the website.

Enabling the discussion directly when the user reads the text is a difficult problem: the integration should not disturb the user, but should encourage her to participate in the discussion. In our first approach we put the discussion system directly between the lines of the article and split the text when the user toggled discuss with a switch. But this slide effect was very confusing and is possibly not usable in most kinds of websites. We then experimented with including the interface below the article. This also proved to be a bad choice since the reader then has to jump to the bottom of the article to participate in a discussion triggered by a statement in the article. In our final version, we used the sidebar to interact with discuss. Optionally, the webmaster can include a second interface by simply adding a `div` with a specific ID.

Using a tooltip can be seen on several websites, like Medium [9]. We added listeners to the article to activate the tooltip, when a text passage has been selected. This provides an unobtrusive method to interact with our application.

The clipboard temporarily stores the user's text selections for later usage. This has been implemented to provide the possibility to read the text, store interesting passages and keep on reading, see Fig. 10. In the end, the user can pick her favorite selection to add it to her argument via drag and drop.

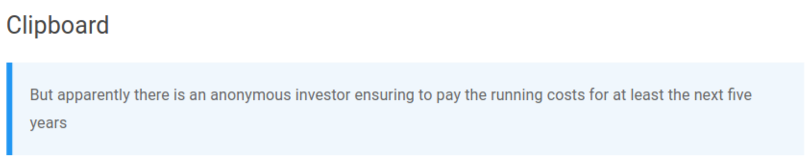


Fig. 10: Using a clipboard to locally store text references.

It is not possible to directly modify the contents of a reference. Our idea is that it should be a direct quotation of the article which is also technically required to find the same text passage in the article. Otherwise, new users will not be able to see the colored reference in the text. We are aware that it is currently still possible to modify the DOM to add a reference of your own desires or to use the browser console for modifications. This would create an untruthful reference, which could lead to false information and false trust in an argument. A server-side verification that the provided string can exactly be found in the article is thinkable, but is currently not implemented.

5.3 Execution Platform

For first testing purposes, we set up a D-BAS instance at our university. As the default configuration uses discuss this backend to directly demonstrate a fully functional application with global discussions enabled. It is possible to use its own backend, which is conform to our application. Therefore, it is not necessarily needed, that the backend is a D-BAS instance – it just needs to provide a suitable interface so that discuss can interact with it.

We are following common best-practices in web development and implemented a RESTful API in D-BAS to expose an interface for external applications, who want to use this dialog-based backend for their applications, whilst discuss is the first project using this interface. This approach for discussion software has already been described in [10] and it presents the general approach how to achieve reusable components in software development, which is why we are also following this structure. Furthermore, [10] proposes the idea to encapsulate the core argumentation logic into an own platform called *Dialog Game Execution Platform* to develop a reusable argumentation core and make it accessible for other applications. In our examples from this paper are we using D-BAS as our default execution platform.

6 Conclusion

Asking the readers to leave a comment below an online news media article is common practice on most websites. But with state-of-the-art comment sections, crowded masses of comments are a typical result. discuss helps to structure discussions and to conduct more productive discourses.

In this paper we used techniques from dialog-based online argumentation to enable our idea of more structured discussions in arbitrary contexts. To achieve this, we implemented discuss as a web application, which follows basic principles of our dialog-based approach and extends discussions by enabling references, global discussions and flexible inclusions into websites.

Feel free to test discuss under `http://cn.hhu.de/discuss` and you are welcome to provide us your feedback.

7 Future Work

We are currently working on more use cases of dialog-based discussions and are evaluating, where our approach could enhance the discourse experience on the Internet. Next, we will extend discuss to support more functions from our backend, e.g. premise groups. In addition, we will evaluate our application in real-world applications and try to cooperate with well-known online news media providers.

Since many people are actively participating in discussions in social networks like Facebook, we will investigate how we can integrate structured discussions into this context. Conceivable are solutions as social bots, which interact with the users based on text messages.

12

References

[1] Klein, M.: Using metrics to enable large-scale deliberation. In: Collective intelligence in organizations: A workshop of the ACM Group 2010 Conference, pp. 103–233 (2010). DOI 10.1177/004057368303900411
[2] Spada, P., Klein, M., Calabretta, R., Iandoli, L., Quinto, I.: A First Step toward Scaling-up Deliberation: Optimizing Large Group E-Deliberation using Argument Maps (2014). DOI 10.13140/RG.2.1.3863.5688
[3] The Guardian: The dark side of guardian comments (2016). URL `https://www.theguardian.com/technology/2016/apr/12/the-dark-side-of-guardian-comments`. Accessed on: 2016-10-14
[4] Krauthoff, T., Baurmann, M., Betz, G., Mauve, M.: Dialog-Based Online Argumentation. Proceedings of the 2016 conference on Computational Models of Argument (COMMA 2016) (2016). DOI 10.3233/978-1-61499-686-6-33
[5] Disqus, Inc.: Disqus - the no.1 way to build an audience on your website (2016). URL `https://disqus.com`. Accessed on: 2016-10-14
[6] rbutr: rbutls, debunkings and counter arguments to misinformation on the internet - think again (2016). URL `https://rbutr.com`. Accessed on: 2016-10-14
[7] Bex, F., Snaith, M., Lawrence, J., Reed, C.: ArguBlogging: An application for the Argument Web. Web Semantics: Science, Services and Agents on the World Wide Web **25**, 9–15 (2014). DOI 10.1016/j.websem.2014.02.002. URL `http://linkinghub.elsevier.com/retrieve/pii/S1570826814000079`
[8] Hoffman, L.R., Maier, N.R.: Quality and acceptance of problem solutions by members of homogeneous and heterogeneous groups. The Journal of Abnormal and Social Psychology **62**(2), 401 (1961). DOI 10.1037/h0044025
[9] A Medium Corporation: Medium (2016). URL `https://medium.com`. Accessed on: 2016-10-16
[10] Bex, F., Lawrence, J., Reed, C.: Generalising argument dialogue with the dialogue game execution platform. Proceedings of the 2014 conference on Computational Models of Argument (COMMA 2014) pp. 141–152 (2014). DOI 10.3233/978-1-61499-436-7-141

4.2 Jebediah — Arguing With a Social Bot

Here, we present our peer-reviewed contribution to the demo session being held as part of the 7th International Conference on Computational Models of Argument 2018, published in Meter et al. 2018a.

Christian Meter, Björn Ebbinghaus and Martin Mauve.

"Jebediah — Arguing With a Social Bot"

In: *Proceedings of the 7th International Conference on Computational Models of Argument*, Volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 467–468, IOS Press.

Summary

In this demonstration, we showcase an alternative frontend and interaction possibility with our argumentation system D-BAS. Jebediah is a social agent, which uses the argumentation logic from D-BAS and enables dialog-based discussions directly in Social Networks. We chose Facebook for the initial integration, because of the huge user base and the different discussion possibilities, e.g. through a chat interface or comment-section-based Facebook posts. Users can interact with Jebediah through a chat interface, asking questions or simply writing commands triggering (through some microservices) the core argumentation logic of D-BAS. The reactions from our DGEP is then being preprocessed and prepared for Facebook's API to present the next argument. To extract the premises and conclusions from the user's input, we used the natural language processing framework *Dialogflow*[1] from Google.

Personal Contribution

The initial idea to enhance dialog-based discussions in Social Networks originates from Christian Meter. Implementing a backend and training the Social Agent in Dialogflow to interact with D-BAS was done by Björn Ebbinghaus. Christian supervised this idea and gave feedback to Björn about the interactions and how it should feel like to speak to a computer engine. Both developed several microservices in the backend to translate between D-BAS' data structure and Facebook.

This short paper was mainly written by Christian Meter with additions by Björn Ebbinghaus in Section 2. Martin Mauve gave improvements to the introduction in Section 1.

Importance and Impact on this Thesis

Jebediah is a proof of concept to showcase that real dialogs can be conducted with our concept of DBOA and that we take the "dialog" literally. Problems occurred with processing the user's

[1] `https://dialogflow.com/`

input, because our social agent did not always understand what the user intended to say. Also, interacting in closed ecosystems, such as Facebook, is very restricting and essential functions to enhance the dialogs between the users, e.g. suggesting arguments or navigation options, are missing and can not be implemented on our own. Therefore, interactions in a literal dialog were successful, but, in general, it felt artificial to interact with our social agent, which is why we do not follow this approach anymore.

September 2018

Jebediah – Arguing With a Social Bot

Christian METER, Björn EBBINGHAUS and Martin MAUVE

Department of Computer Science, University of Düsseldorf, Germany
firstname.lastname@hhu.de

Abstract. In this demonstration we will showcase Jebediah, a social bot based on Google's Dialogflow. Jebediah is a front-end to dialog-game execution platforms that enable their seamless integration into popular social networks such as Facebook or Twitter. Users can interact with the social bot using natural language while Jebediah translates the user input to a format that can be interpreted by a dialog game execution platform and vice versa.

Keywords. online argumentation, dialog-game execution platform, artificial intelligence, social networks

1. Introduction

In prior work we introduced the *Dialog-Based Argumentation System* (D-BAS)[1] [1], a *Dialog Game Execution Platform* (DGEP) [2] for dialog-based online argumentation. D-BAS allows users to exchange proposals and arguments with each other in the form of a time-shifted dialog where arguments are presented and acted upon one-at-a-time. It is designed as a full-stack, stand alone web-application.

However, currently, the vast majority of online discussions takes place in social networks such as Facebook or Twitter and not on dedicated argumentation web-sites. We therefore investigated how the functionality of a dialog-game execution platform, such as D-BAS, can be included in a seamless way into social networks. Our solution to the problem is a social bot called Jebediah. It provides a front-end to DGEPs that can be integrated into social networks in a seamless way.

2. Jebediah – a social bot for online argumentation

Jebediah is a social bot based on Google's framework *Dialogflow*[2] for *Artificial Intelligence* (AI) development. It connects Dialogflow with a DGEP such as D-BAS' backend. Dialogflow enables a seamless integration into many popular social networks, e.g. Facebook or Twitter, and provides processing of text-input from conversations. We leverage this to enable natural language access where the AI is used to parse and interpret the user's input, whereas the interpreted data is sent to D-BAS' DGEP, in order to calculate the next steps in the discussion. This setup allows us to directly have a conversation with

[1] `https://dbas.cs.uni-duesseldorf.de`
[2] `https://dialogflow.com`

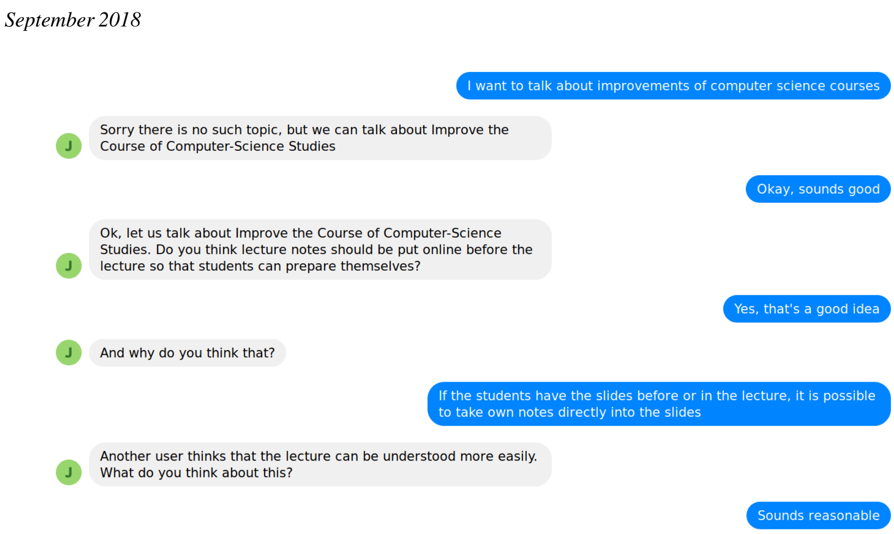


Figure 1. Conversation with Jebediah. On the right side is the user's input. On the left side are the answers.

interested users without the need to leave the current site and to provide a solution to reduce crowded comment sections, e.g. inside a Facebook post.

Jebediah exposes the full functionality of D-BAS, i.e. collect statements from users, integrate them into a discussion graph and present the next statement to the user (see Fig. 1). It is then possible, in natural language, to interact with arguments and experiences from those users. Users can also start a dialog with the agent and ask for possible entities in the discussion, e.g. topics or other positions.

Where D-BAS' interface shows the user a list of possible steps to choose from, this is hardly manageable in a text-only or even voice-only environment. Therefore Jebediah has to lead the user in a way that advances the conversation into deeper levels of the topic, while being flexible enough to react to user actions which are not a usual part of the D-BAS discussion flow. This is even more important in a voice interface where the user has to memorize the current part of the discussion.

3. Related Work

Arvina [3] is a system that bears a lot of similarities to our work. With Arvina it is possible to replay previously stored discussions and interact with the recording. Multiple real users can participate in the debate and also add new statements. Jebediah, in contrast, aims at enabling a seamless integration of a DGEP into social networks and at providing a discussion using natural language.

References

[1] T. Krauthoff, C. Meter, G. Betz, M. Baurmann, and M. Mauve, "D-BAS – A Dialog-Based Online Argumentation System," in *Computational Models of Argument*, September 2018, pp. 325–336.
[2] F. Bex, J. Lawrence, and C. Reed, "Generalising argument dialogue with the dialogue game execution platform." in *COMMA*, 2014, pp. 141–152.
[3] M. Snaith, J. Lawrence, and C. Reed, "Mixed initiative argument in public deliberation," *Online Deliberation*, 2010.

4.3 discuss vs. Disqus: Evaluating Dialog-Based Discussions Against a Comment-Based System

With the goal of gathering feedback about our software stack, we conduct a study testing our stack of discuss, D-BAS and EDEN and compare it against a common comment section tool in Meter et al. 2020:

Christian Meter, Alexander Schneider, Marc Feger, Jan Steimann and Martin Mauve.

"discuss vs. Disqus: Evaluating Dialog-Based Discussions Against a Comment-Based System"

Manuscript submitted for publication.

Summary

In order to test our stack of software tools encouraging dialog-based discussions, we conducted a study in which we evaluated our DBOA approach against a common comment section tool. We prepared three online articles about a vegetarian diet and asked students from our university to voluntarily participate. These articles were prepared in a way that it was either possible to discuss in a dialog-based fashion with discuss or with the classical approach of the comment-section-based system Disqus.

In total 62 students participated, were organized in groups and then either presented with discuss or Disqus as their discussion-frontend. We observed that the group using discuss produced more than twice as many arguments than the Disqus-group, which is a significant improvement on the outcome from a discussion. We counted the number of arguments independently in a team of four researchers, namely the first four authors of this paper. Counting the arguments led to very different results between the four annotators, considering that it is a difficult task to understand and extract all arguments packed into a running text, which is the output encouraged by Disqus. Therefore, the discrepancy regarding the number of counted arguments was much higher compared to arguments collected through discuss.

The study was brought to a close with a survey, comparing both tools regarding the usability and the overview they provided concerning the discussion. The results show that discuss performs as well as Disqus in terms of comprehensive use in a discussion, which is a great success for us. But we recognized problems with the design of our user interface providing an inferior user experience compared to established software.

Personal Contribution

Designing and conducting the experiment was jointly performed by the author of this thesis and Alexander Schneider. Most parts of the paper were written by both authors, with a focus on the project setup, experiment design and hypotheses by Christian and a focus on the results and interpretations of the results by Alexander. Marc Feger and Jan Steimann supported the experiment and provided further analysis of the results in Subsection 4.3 "Lessons from Annotator Differences". Martin Mauve contributed to the experiment design and gave general feedback on the sections.

Importance and Impact on this Thesis

Conducting a study in which untrained users tested our software stack is an essential contribution to the future of this project. The study showed that our approach is competetive and comparable to established software. Therefore, despite needing some optimizations, our approach remains the same, partly for the reason of encouraging users to deliver a greater number arguments. With the encouragement from the study, we are going to keep developing the DBOA-approach to improve the user experience which we will follow up with another field study to compare the changes.

Important Note

This manuscript is currently submitted for publication.

September 2020

discuss vs. Disqus: Evaluating Dialog-Based Discussions Against a Comment-Based System

Christian METER [1], Alexander SCHNEIDER[1], Jan STEIMANN, Marc FEGER and Martin MAUVE

Computer Networks Department
Heinrich-Heine-University Düsseldorf, Germany
firstname.lastname@hhu.de

Abstract. In this work we present the results of a hypotheses-guided lab experiment comparing the discussion of online newspaper articles by means of regular comment sections versus the use of our own, dialog-based approach. We show that the main problem of our approach is usability and user interface design. At the same time we can prove that it has a large positive impact on the number and clarity of users' arguments. As a consequence we reason that more effort should be spent on user interface and user experience design of systems that support online argumentation.

Keywords. argumentation, argumentation system, online discussion, dialog-based, web-application, study

1. Introduction

The focus of research on online argumentation, so far, has mainly been on either theory or on designing novel systems. Some of those systems have then been put to the test by using them in lab or real-world settings. Typically, the authors of those systems report that the tests were quite successful. However, at the same time, the collective research in this area has had limited impact on how online discussions and argumentations are conducted in the real world. In the vast majority of real-world applications, some form of forum- or comment-based system is still used. An approach that our research community thinks of as being deeply flawed.

In an attempt to shed some light on why that might be the case, we have conducted a hypothesis-guided lab experiment. In this experiment we compared discuss [1], our own approach to support online argumentation, with Disqus[2], a commonly used comment system.

The main findings presented here are as follows. First, we provide very detailed information regarding the advantages and drawbacks of using discuss in comparison to

[1] Both authors contributed in equal parts to this work.
[2] `https://disqus.com`

Disqus. Second, we demonstrate that hypothesis-guided lab experiments provide important insights. Finally, the results of our experiment indicate that our approach has significant potential to outperform forum based systems. However, it is held back because participants are not familiar with our system and the user interface is not good enough to compensate for this.

The remainder of the paper is structured as follows: Section 2 describes the experiment setup, our research questions and hypotheses. In Section 3 we present the data gathered through the experiment and a statistical evaluation of the results. Following, Section 4 contains our interpretation of the results. Related work is discussed in Section 5. We conclude the paper with a summary and an outlook in Section 6.

2. Experiment

2.1. Argumentation Software

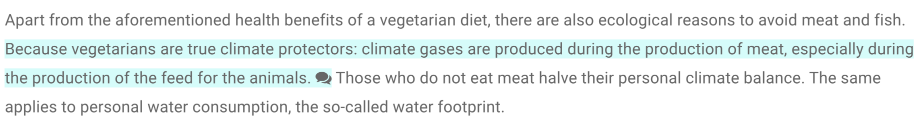

Apart from the aforementioned health benefits of a vegetarian diet, there are also ecological reasons to avoid meat and fish. Because vegetarians are true climate protectors: climate gases are produced during the production of meat, especially during the production of the feed for the animals. Those who do not eat meat halve their personal climate balance. The same applies to personal water consumption, the so-called water footprint.

Figure 1. A text reference created with discuss. Clicking on the highlighted part, jumps into the discussion shown in Figure 2.

We used two different software tools in our study. The first is discuss, our own tool for dialog-based online discussions. In discuss users can mark a section of a web page and attach an argument to it. As shown in Figure 1 this section becomes highlighted and other users can click on it to see the attached arguments and enter a dialog-based discussion.

In dialog-based discussions the user is shown an argument of another user and can react to it. This is depicted in Figure 2. One possible reaction is to attach another argument. Based on the reaction the user is then confronted with the next argument. In this way the user conducts a dialog with the system, while the system represents all users that have already added arguments in the past.

The other software is Disqus, a popular tool to embed hosted comment sections into websites. Users can add their comments and reply to others, see Figure 3. Disqus was used because it provides a similar feature-set to discuss, e.g. inter-article discussions, which makes it a good comparison.

2.2. Research Hypotheses

The main goal of our study was to get a good understanding whether users would accept or possibly even prefer discuss as a replacement for common commenting tools such as Disqus. To this end we set up the following series of hypotheses before conducting the experiment.

H1 *Using "discuss" is as intuitive as using "Disqus".*

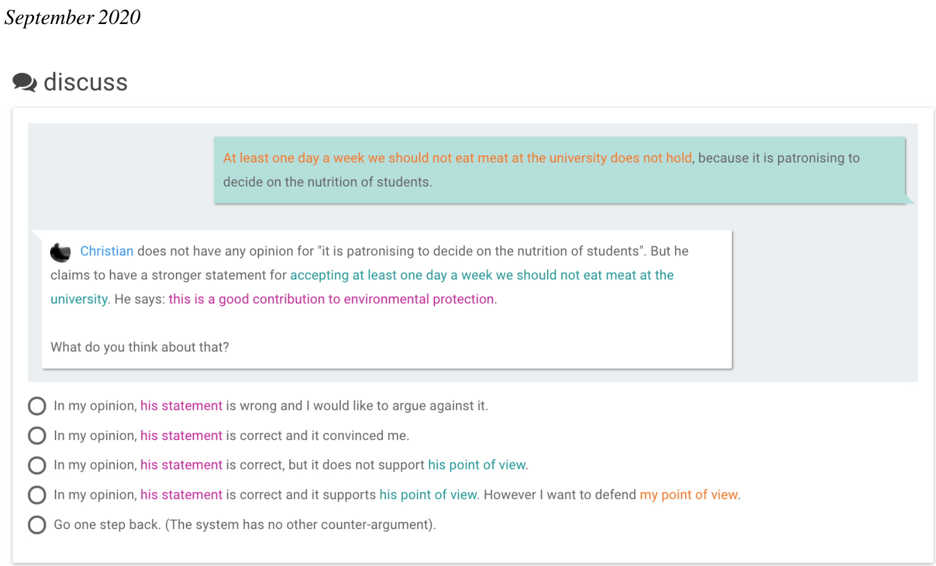


Figure 2. A reaction-step in a discussion with discuss.

H2 *"Discuss" offers the necessary flexibility to comment on a specific aspect of the article.*

H3 *Other users' contributions are of interest to the user.*

H4 *It is easy for users to get used to the "discuss" user interface.*

H5 *With "discuss" it is not easier to understand the context of an argument made by another participant.*

H6 *With "discuss" it is not easier to gain a good overview of a discussion.*

H7 *It is not helpful to use "discuss" to argue across articles.*

H8 *The ability to reuse arguments is used more frequently with "discuss".*

The reason why we used a mixture of positive and negative (in relation to discuss) hypotheses is that we generally tried to formulate the hypotheses in a way as to be able to disprove them in a statistically significant way and at the same time learn how to best proceed with the development of discuss.

H1 and H4 aim at measuring the subjective feeling of the users to compare discuss and Disqus regarding accessibility. This was important to us since we anticipated that our own tool might have problems in this area because we are no experts in user interface design.

Discuss allows users to directly interact with the text of the articles. We therefore expect it to do better in regard to commenting on one specific aspect of the article. This is captured by H2. We expect many interactions with passages in the article and therefore a good result when evaluating this hypothesis for discuss — at least better than Disqus.

H3 targets the general interest in the opinion of other users. Since the users are participating in the discussion voluntarily, we are expecting both groups to have a high interest in the topic and the opinions of other users.

By asking for the context of an argument, we are expecting in H5 that the argument's context is clearer in discuss than in regular comment sections. H6 is set up to prove a

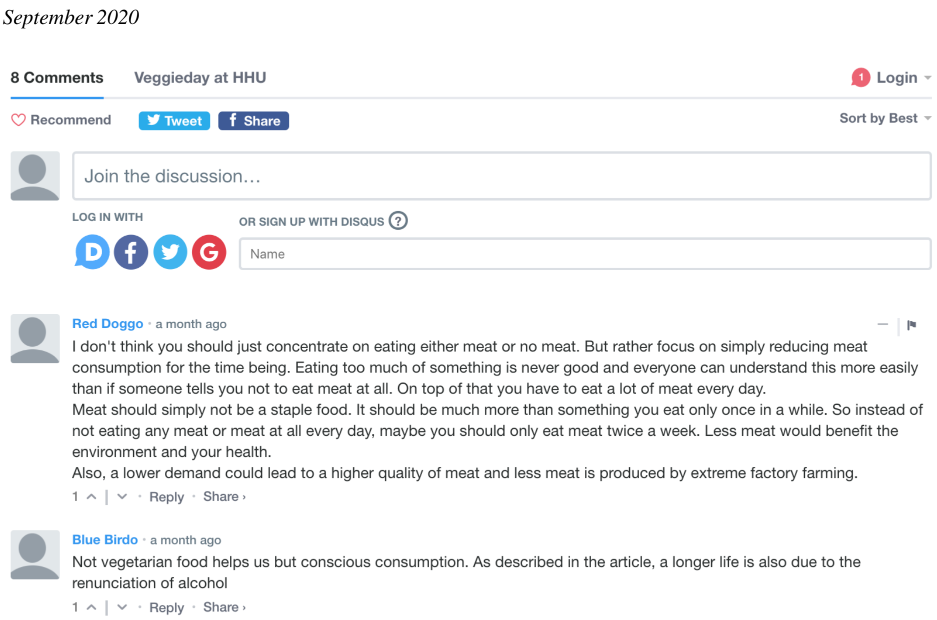


Figure 3. Two sample comments of the discussion with Disqus.

similar result, by targeting an overview of the complete discussion. We anticipated that discuss should perform better than Disqus in this area as well.

H7 refers to the mechanism for inter-article discussions. Both groups have the technical means to use this feature. We expect better results for discuss, because of a deeper software-integration of this feature.

Re-using arguments to reduce redundancy is one of the key goals in the dialog-based discussion approach. Thus, discuss has mechanics implemented to reuse previously submitted arguments, which are covered by H8. We expect at least some amount of arguments to be reused as the discussions develop.

2.3. Experimental Procedure

Students in groups from four to eight people were invited to join the discussion in person at one of our computer labs (see more details in Section 3.1). The computers were prepared to have a clean browser and three online articles about vegetarian diet opened. These articles either integrated discuss or Disqus. Participants using Disqus are from this point on referred to as the *control-group*. All browsers were opened so that users could directly participate in the discussion. The initial state of the discussion contained two arguments provided by us as a starting point.

Each participant had an own computer with random credentials for participation. It was not possible for them to look on the screen of other users, and they were instructed to only communicate online. A text-tutorial was attached to their screens to explain in a few words how to use the tools. The discussions were saved and reset to the initial state after each run.

Both tools allowed reading arguments of other users and to add new ones. The control-group was able to discuss in the typical comment-reply pattern as it is broadly used in online news media. Discuss users were guided through the discussion as it is known of *Dialog-Based Online Argumentation [2,3]* (DBOA).

September 2020

Participants were allowed to browse the Internet freely during the experiment, e.g. to acquire background information or to look up facts. We did not moderate any of the arguments and did not participate in the discussion. Only technical support was given by us if something was unclear.

Each group of participants was first instructed about the procedure of the study with the exact same text read aloud by one of the authors. Afterwards, they had 30 minutes to discuss, ten minutes to answer the questionnaire and were in the end awarded with € 10.

We announced the study in several lectures, posted flyers on bulletin boards, posted on Twitter and came into direct contact with the students on our campus. All of them participated freely in the discussion. The participants could choose between eleven dates, all taking place within three weeks. A twelfth date that we provided to even the number between the groups did not get any reservations.

3. Results

We obtained two kinds of results from the experiment. The first kind are subjective ratings from the participants, regarding their perceptions about the software they used and the discussion they led. For this the participants were presented with a questionnaire containing assertions, which they had to rate on a five-part *Likert Scale* [4], ranging from one, representing "I absolutely disagree" to five, representing "I absolutely agree". The questions and the results are shown in Table 1. We used the Mann-Whitney-U-Test to test for statistical significance of the differences in rating. We further reject or accept our hypotheses based on those results that are significant. A part of the questionnaire was only answered by users in the discuss-group, since the questions targeted properties of discuss specifically. The second kind of data is the data produced directly by the users, for example the number and content of arguments. We annotated the statements that the participants produced with four annotators and compared, e.g., the number of arguments per statement between the control- and the discuss-group.

#	Question	Average		Median		Variance		MMW
		discuss	control	discuss	control	discuss	control	
1	I was personally interested in the topic	4.229	4.148	4	4	0.462	0.8669	$p: 0.9876$
2	I would participate in a discussion for a similar topic	4.4	4.385	5	5	0.5257	0.7751	$p: 0.8253$
3	I understood how to participate in the discussion	4.086	4.692	4	5	0.8784	0.5207	$p: 0.0041$
4	It was easy to comment on a specific part of the article	3.667	4	4	4	1.434	0.963	$p: 0.3293$
5	The comments of other users interested me	4.235	4.296	5	5	0.7682	0.8752	$p: 0.6772$
6	I understood how the discussion worked	3.647	4.63	4	5	1.287	0.4554	$p: 0.0005$
7	I had the feeling that a lot of the comments did not fit the topic	2.429	1.538	2	1	1.445	0.7101	$p: 0.0019$
8	I had the feeling that I had a good overview of the discussion	2.771	3.481	3	4	1.319	0.7682	$p: 0.0154$
9	I think that multiple articles for the same topic enriched the discussion	3.909	4.346	4	5	1.355	0.8417	$p: 0.1507$
10	I learned something through the comments of other participants	3.086	2.808	3	2	1.678	1.386	$p: 0.3936$
11	I gained a new perspective regarding the topic through the discussion	2.657	2.519	3	2	1.425	1.805	$p: 0.5895$
12	I lost track of the content of the discussion	3.029	1.846	3	1	1.628	0.9763	$p: 0.0005$
13	The participants treated each other respectfully	4.781	4.444	5	5	0.2334	0.8395	$p: 0.1561$
14	Highlighting sentences inside the article was disruptive	1.857	-	2	-	1.094	-	-
15	The suggestion of arguments was helpful	2.853	-	3	-	1.831	-	-
16	The tool "discuss" helped the discussion	3.212	-	3	-	1.379	-	-
17	I understood how to navigate through the discussion	3.6	-	4	-	1.154	-	-
18	"discuss" enables better discussions than traditional comment boxes	3.125	-	3	-	1.234	-	-
19	I think that "discuss" leads to a more intense reflection of the arguments	3.871	-	4	-	1.209	-	-
20	"discuss" is too complicated and I got lost	2.5	-	3	-	1.132	-	-
21	I think "discuss" leads to a more respectful discussion between the participants	3.267	-	4	-	1.596	-	-

Table 1. Translations of the questions from the survey the participants had to fill out after the discussion. We used a 5-point Likert scale for each question, ranging from 1: "I absolutely disagree" to 5: "I absolutely agree".

In the following, we highlight the differences between the two groups, discuss whether those differences are statistically significant and in cases where they relate to our hypotheses, whether they can be used to confirm or reject them.

3.1. Participant Data and Composition

Overall, 62 students participated in the experiment in 11 groups. Participants were allocated to five control-groups with 27 members in total, while 35 students were presented with the discuss-software in six groups. The age distribution ranged between 17 and 50 with a mean of 23.17 for the control-group and 23.68 for the discuss-group. In the control-group 15 participants were female, 10 male, one person chose “not specified”, and one did not fill out the corresponding form. For the discuss-group, 18 participants were female, 16 male and one chose “not specified”. Control-group participants were from 13 different degree courses and had a semester average of 3.889 while the discuss-group participants were from 16 different degree courses and averaged 5.057 semesters.

3.2. Discussion Quality Perception

We used the first set of questions (1-7) from Table 1, which were answered by participants in both the control- and the discuss-group, to measure how they perceived the quality of the discussion itself. At first, Questions 1 and 2 asked the participants for their interest in the topic. Our results showed no difference between both groups, which implies that a predisposition to topic preference did not color the following results.

Question 3 (“I understood how to participate in the discussion”) was supposed to test for the intuitive usability of the software. Here discuss had a worse, statistically significant, outcome than the control-group software. This directly disproved H1, which was expected since the participants are used to comment-boxes and for the most part heard the first time about dialog-based argumentation during the experiment. Similarly, the results from Question 6 disprove H4 as well.

H2, and H3 on the other hand held, as denoted by the results from Questions 4 and 5. Incidentally, both of the supported hypotheses target inherent qualities of discuss. It was assumed that discuss would at least perform equally to conventional comment-boxes, which it did.

Participants that used discuss felt more strongly that the comments of others were unfitting, which is shown by Question 7, in turn implying that H5 does hold.

3.3. Overview of the Discussion

A second set of questions (8-13) tested if the participants were able to navigate the discussion or whether they could gain a rough idea what the discussion was about.

Two of the questions belonged together and should have a related outcome to gauge whether the participants where answering thoroughly or just clicked randomly. Those were Questions 8 and 12 as well as 10 and 11. In all cases the results did fit.

For Questions 8 and 12 the results for discuss were worse and therefore supported H6. Questions 10 and 11 seem to support this further, although the differences are smaller and not statistically significant. The ability to comment on several articles with the same software was queried by Question 9. Even though the results are slightly in favor of the control-software, they are not statistically significant and thus H7 is rejected.

September 2020

Additionally, we presented the statement "*The participants treated each other respectfully*" which was not linked to a hypothesis and was included to gain a sense whether the participants felt respected. A difference of 0.34 in favor of discuss was not statistically significant ($p = 0.1562$).

3.4. *Discuss-Specific Questions*

The last set of questions (14-21) was only presented to participants in the discuss-group, as they reference certain features of discuss, which are not directly comparable to the comment software used with the control-group.

Results suggest that H8 does not hold, since the according survey question was on average answered with 2.853 points and a variance of 1.831. As a reminder: 1 represented that the participant felt no use at all for the argument suggestions while 5 represented that the participant felt the suggestions were very helpful. Other results in this section showed mediocre outcomes for features of discuss, except for a perceived heightened sense of critical thinking in regard to the arguments of the discussion in Question 19. Five of the participants in the discuss-groups acknowledged that they at least heard of the dialog-based argumentation style before. The other 30 participants were confronted for the first time with dialog-based discussions.

3.5. *Content Difference*

Besides the questionnaire answers, we also analyzed comments produced by the participants. Looking at the number of "statements", a user produced *4.88* on average for the control-group and *8.26* for the discuss-group. A "statement" is a typical comment, not regarding whether it contains an argument or not. It is important to state that in a lot of cases the discuss-software, by design, forces the user to enter two "statements". This is, for example, the case when the user adds a new argument and needs to provide at least one statement for the premise and one for the conclusion. 52.67% of the statements in the control-group were a direct reaction to the statement of another user, which suggests a high interactivity. The rate for the discuss-group is 100% and not comparable, because participants react to the statements of other users by design.

To gauge the total number of produced arguments, we used four annotators that worked through the statements and noted the number of arguments contained in them. A nonrestrictive definition was used to define an argument: It needed to contain at least one premise and a conclusion. We measured the inter-coder reliability through the Holsti method. The overall reliability was 76.29%, which is usually on the brink of acceptability for argument-annotation from natural text. When we look at the *Holsti Index* [5] for the statements from the control- and discuss-groups separately, we get a reliability of 55.47% and 88.96% respectively.

Depending on whether we take the lowest, highest or the average scores produced by the annotators, 247, 418 or 330 arguments have been produced in total across all groups, respectively. This means on average every participant produced 5.24 arguments. The spread of possible arguments (and thus the disagreement between the annotators) is significantly higher when only the control-groups are considered. Then we get 27 (lowest), 149 (highest) or 94 (average) arguments, with 3.48 arguments per participant. In contrast, the discuss-groups, which had 8 participants more, produced 220 (lowest), 269 (highest) or 236 (average) arguments, which results in 6.75 arguments per participant. Further analysis and explanation of this disparity is given in Section 4.

4. Discussion

The questionnaire data in itself produced mixed results regarding the previously stated hypotheses. Since not all have been rejected, we will now discuss conclusions that can be drawn.

4.1. Questionnaire Implications

One key result of the evaluation is that the hypotheses concerning the intuitive handling and usability of discuss (H1 and H4) were rejected and that it was not easier to understand the context of an argument (i.e., H5 was accepted). This result is important since it is very unlikely that a system with these problems will achieve widespread use in real-world environments. We believe that there are three main reasons for this outcome. First, participants are used to existing forum-based systems, therefore those systems have an implicit advantage regarding usability. Second, entering and interacting with arguments is likely harder than just writing and referring to plain text. Third, we are no experts in user experience and user interface design, thus it is very likely that both user experience and user interface are far from being optimal. Out of those three reasons only the last one can be changed. Improving the user interface and the user experience will therefore gain a very high priority in our future work.

Discuss got significantly worse grades for the statement "I had the feeling that a lot of the comments did not fit the topic", which supported H5. We expected scores to be better or at least equal to the control-group, since we assumed, that discuss enforces a more factual discussion with less off-topic comments. One possible explanation is that users are always confronted with a counter-argument to their last statement. Continuing the discussion this way, could produce a "rabbit-hole" effect, whereby the user is debating increasingly irrelevant seeming sub-issues. In contrast, using comment-boxes allows the user to see several comments at once and thus pick the more fitting ones. It was also easier in the control-group to keep the overview because of the lower number of arguments, which were all produced during the experiment. Therefore, the participants could keep track of all changes in the discussion, which is not always the case in bigger discussions.

An interesting observation is that users seem to perceive the participants of the discuss-version to be more respectful towards their peers. Although, this may only be a trend since the differences could not be proven to be statistically significant ($p = 0.1561$). This would be plausible, since discuss enforces a more strict argument-focused style of discussion, which causes the participants to use less *ad hominem* and other uncalled-for behavior. Users interact more on an argument-centric interface, instead of the typically personal message-based interface of comment-boxes.

Participants mostly agreed that discuss leads to a more intense reflection of arguments, which again makes sense since discuss focuses on arguments rather than personal opinion. Other interesting results were that most users did not feel disturbed by highlighting parts inside the article. This knowledge can be used in the future to strengthen

September 2020

objectivity in arguments by enabling the user to mark sentences as a kind of citation or direct reference. A similar observation has already been made by Mullick et al. [6]. They conducted experiments to test the hypothesis, that users mainly do not read the whole article and mostly comment on specific parts. Out of the 20 participants of this study 17 stated that they enjoyed the possibility to comment specific paragraphs instead of the whole article.

4.2. Produced Arguments

The most important result from the analysis of the content produced by the participants was the difference in the number of statements and arguments per participant. The participants in the discuss-group produce more than double the amount of arguments per user, which was to be expected as an outcome of an argument-focused interface.

Furthermore, it was also more clear-cut for the annotators what statements constituted one or more arguments. The *Inter-Annotator Agreement [7]* (IAA) between any two annotators was between 84.47% and 95.43% for statements from the discuss-group, while it was between 42.75% and 70.23% for the control-group. This hints at two things: Firstly, statements produced by discuss seem to contain more clearly structured arguments. Secondly, statements given by plain text commenting on an article seldom contain clearly identifiable arguments.

We believe that the main reason for this is that a lot of the comments produced by users in the control-group were written in a way, that did not state their intentions explicitly, but more or less implied what they wanted to express. This has the effect that it depends on the reader and their current state of mind whether they register the possible arguments contained in the comment. Another reason could be that when the users are not nudged towards producing arguments, they simply only state their opinions or any kind of off-topic comments without the intention of starting or participating in a conversation or debate. Related studies show this as well: When presented with traditional commenting options users often participate for the purpose of asking questions, provoking others, providing new perspectives [8] or for purely entertainment value [9].

4.3. Lessons from Annotator Differences

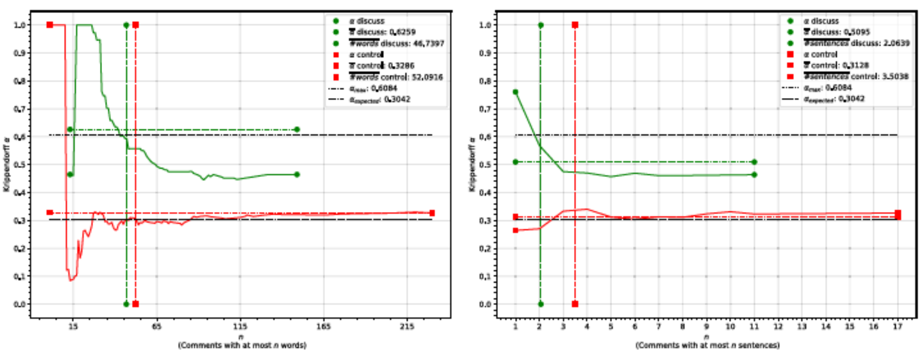


Figure 4. Development of Krippendorff α against the number of words and sentences in the comments.

Additionally, the systems can be compared using the argument annotations and the resulting IAA. For this purpose, we have annotated each of the 350 comments with the corresponding number of arguments. In order to not distort the annotation and to preserve the individual view of the annotators, no correction phase was carried out. As an additional IAA measure, we used the study-specific Krippendorff α in combination with the ratio-distance function [10,11]. Overall, a match of $\alpha_{global} \approx 0.535$ was achieved on the data of both groups. The highest result with $\alpha_{max} \approx 0.608$ was achieved by omitting an annotator. Accordingly, the lowest result was also achieved with $\alpha_{min} \approx 0.477$. In the control-group, an agreement of $\alpha_{control} \approx 0.327$ for all annotators with a total of 131 comments was observed. Analyzing the 219 comments obtained by discuss, $\alpha_{discuss} \approx 0.464$ could be measured. A closer look at the data showed that a major problem in the implementation of the annotation is the number of sentences and words used within a comment. About 25% of the comments had plenty of long sentences. On the remaining 75% of the data a global agreement of $\alpha'_{global} \approx 0.6$ could be measured. For the two groups the agreement with $\alpha'_{control} \approx 0.315$ as well as $\alpha'_{discuss} \approx 0.524$ could thus be established. Despite a correction phase, small IAA values and high complexity, similar outcomes were determined as sufficiently good by [12] for a comparable annotation task.

Figure 4 shows the development of the IAA value on the non-adjusted data. All comments with a minimum number of words or sentences are examined. Both figures show that α falls with an increasing number of words or sentences per comment. It is clear that the IAA is high until more comments with plenty of long sentences are included. Exceeding the word and sentence boundaries leads to a divergence in the annotators views. While the agreement for the control-group oscillates around the expected value $\alpha_{expected} \approx 0.3$, it is clear that the data generated in discuss always produces a high degree of agreement regarding the recognition of the arguments it contains. Since discuss specifies a pattern for the input, arguments and their structure can be better recognized. In comparison with the control-group, which does not provide such a structure, the results obtained by discuss are better with regard to α. Therefore, the decrease of the α value with respect to discuss could be explained by the fact that by adding more long sentences it is no longer possible to distinguish between the different forms of arguments as described by [13]. Nevertheless, it turns out that the structure is essentially involved in the interpretation of arguments. Thus, a dialog-based system, like discuss, supports this understanding better than a simple system.

5. Related Work

Several experiments researching effects of online discussions have been conducted. Lampe et al. [14] researched how civility in online discussions is affected by choice of moderation system. They used the forums of *Slashdot* as a control for civil discussions and moderation. Another field-study by Rhee and Kim [15] tested whether online discussions could change the quality of a deliberative process. However, they conducted their experiments on the Internet and not in a lab setting. A study similar to this paper was conducted by Iandoli et al. [16]. They pit their collaborative online discussion tool against conventional forums, which are threaded and comment based. Here an online political process of an Italian party was used instead of a controlled lab setting.

Other studies utilized dialog-based discussions in their experiments as well. Krauthoff et al. [17] conducted a study where more than 300 students participated online in a discussion regarding the betterment of a study course. In contrast to this work no control-group was used. Another experiment utilizing dialog-based discussions was done by Ebbinghaus [18,19]. In this case the test was more geared toward whether decision-making processes based on dialog-based discussions are viable.

6. Conclusion

In this paper we presented a hypotheses-guided lab experiment to compare traditional comment-style online-argumentation and a dialog-based alternative. The main results show that participants prefer using the traditional comment-style methods and are overall more accustomed to them. We argued that improving the design of the user interface and improving the user experience should be a top priority in order to change this. As a second key result our data shows that the dialog-based approach leads the participants to produce both significantly more arguments and better structured arguments. It is therefore clearly worthwhile to put effort into improving the user interface and the user experience.

A secondary result is that experiments, as described in this paper, are a valuable tool to understand what the real-world problems and benefits of a proposed online argumentation approach are. In order to concentrate our efforts on the bottleneck issues and not on side issues it makes a lot of sense to use them more often. In the following we therefore outline how to improve the experiments themselves.

Since our participants were predominantly students, one could assume a certain bias towards discussion affinity. One way to improve the results would therefore be to conduct tests with subjects that conform to an intersection of the general populace. Furthermore, repeating the study with different articles and the same participants could give us a hint whether familiarization with the software might increase the usability ratings.

Another aspect that might be worthwhile to look into more detail is the annotation of arguments contained in the comments. We followed a very open definition since no participant was trained in formal argumentation. It would be interesting to see how the annotations change, when more restrictive definitions of argument are used.

It would also be very interesting to repeat the study with slight variations to determine the impact of those variations on the outcome. For example, using a dialog-based system with a different interface than discuss could show whether the lower usability ratings are inherent to dialog-based approaches or are caused by specific implementation details. A repeat-study with a less controversial topic could help to understand whether comment based approaches fare better, when there is less need for argumentation.

Acknowledgements Thanks to Dennis Frieß and his support to prepare the study. He also pointed us to some papers giving us a better understanding on conducting studies.

September 2020

References

[1] C. Meter, T. Krauthoff, and M. Mauve, "discuss: Embedding dialog-based Discussions into Websites," in *Human Computer Interaction International: Proceedings of HCI*, 2017.

[2] T. Krauthoff, C. Meter, G. Betz, M. Baurmann, and M. Mauve, "D-BAS – A Dialog-Based Online Argumentation System," in *Computational Models of Argument*, September 2018, pp. 325–336.

[3] A. Schneider and C. Meter, "Various efforts of enhancing real world online discussions," in *ECA 2019: Proceedings of the 3rd European Conference on Argumentation*, June 2019.

[4] R. Likert, "A technique for the measurement of attitudes." *Archives of psychology*, 1932.

[5] O. R. Holsti, "Content analysis for the social sciences and humanities," *Reading. MA: Addison-Wesley (content analysis)*, 1969.

[6] A. Mullick, S. Ghosh, R. Dutt, A. Ghosh, and A. Chakraborty, "Public sphere 2.0: Targeted commenting in online news media," in *European Conference on Information Retrieval*. Springer, 2019, pp. 180–187.

[7] R. Artstein, "Inter-annotator agreement," in *Handbook of Linguistic Annotation*. Springer Netherlands, 06 2017, pp. 297–313.

[8] M. Ziegele, T. Breiner, and O. Quiring, "What creates interactivity in online news discussions? an exploratory analysis of discussion factors in user comments on news items," *Journal of Communication*, vol. 64, no. 6, pp. 1111–1138, 2014.

[9] N. Springer, I. Engelmann, and C. Pfaffinger, "User comments: Motives and inhibitors to write and read," *Information, Communication & Society*, vol. 18, no. 7, pp. 798–815, 2015.

[10] K. Krippendorff, "Content analysis: An introduction to its methodology." Sage Publications, 2004, vol. 2, pp. 241–s243.

[11] ——, "Computing Krippendorff's Alpha-Reliability," https://repository.upenn.edu/asc_papers/43/, 2008, accessed: 2020-02-21.

[12] F. Boltuzic and J. Šnajder, "Back up your stance: Recognizing arguments in online discussions," 01 2014, pp. 49–58.

[13] J. Lawrence and C. Reed, "Argument mining: A survey," *Computational Linguistics*, pp. 1–55, 10 2019.

[14] C. Lampe, P. Zube, J. Lee, C. H. Park, and E. Johnston, "Crowdsourcing civility: A natural experiment examining the effects of distributed moderation in online forums," *Government Information Quarterly*, vol. 31, no. 2, pp. 317–326, 2014.

[15] J. W. Rhee and E.-m. Kim, "Deliberation on the net: Lessons from a field experiment," *Online deliberation: Design, research, and practice*, pp. 223–232, 2009.

[16] L. Iandoli, I. Quinto, P. Spada, M. Klein, and R. Calabretta, "Supporting argumentation in online political debate: Evidence from an experiment of collective deliberation," *new media & society*, vol. 20, no. 4, pp. 1320–1341, 2018.

[17] T. Krauthoff, C. Meter, and M. Mauve, "Dialog-Based Online Argumentation: Findings from a Field Experiment," in *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence*, vol. 2012. CEUR-WS, 2017, pp. 85–99.

[18] B. Ebbinghaus, "Decision Making with Argumentation Graphs," Master's thesis, Department of Computer Science, Heinrich-Heine-University Düsseldorf, May 2019.

[19] B. Ebbinghaus and M. Mauve, "decide: Supporting participatory budgeting with online argumentation," in *Submitted to Computational Models of Argument*, 2020, under review.

Chapter 5

Arguments as a Resource

In these last chapters, we present our research findings and solutions to some research questions, e.g. making argumentation theory accessible to untrained users, enhancing the discussion experience on the Internet and introducing structure into discussions. To reduce redundant arguments, we have to go one step further, leading us to a point where we need to think about the way we define an argument.

An argument is currently defined as a 3-tuple consisting of a conclusion, a relation and (multiple) premises. At present, arguments are ephemeral data, even if dialog-based discussion systems are being used. In order to proceed, we have to *reuse* arguments and even whole argumentational structures.

Either an exchange of arguments between content providers obtaining those from a central service, where arguments are stored, occurs naturally as an idea. We have seen similar approaches in the Argument Web (Bex et al. 2013; Rahwan et al. 2007) and the AIFdb (Lawrence et al. 2012b), which collects complete discussions. Splitting these up into smaller building blocks, i.e. into the separate parts of an argument, is a better prerequisite to enable reusing arguments in different discussions.

Therefore, in the next sections we propose the general idea of interpreting arguments as valuable data instead of simple text entries. This data should be exchanged between content providers, to present it to the users that are discussing their content, enabling others to use the data arising here in turn. Furthermore, we describe an update-mechanism for "old" arguments, which should be revised, similar to version control systems in software development. These ideas are then implemented in the EDEN project, enabling the exchange of arguments and facilitating their reuse via a discuss-interface.

5.1 Reusable Statements in Dialog-Based Argumentation Systems

In this section we present the findings and first ideas concerning a distributed system of arguments, which was published in Schneider and Meter 2017:

Alexander Schneider and Christian Meter.

"Reusable Statements in Dialog-Based Argumentation Systems"

In: *Proceedings of the 1*st *Workshop on Advances in Argumentation in Artificial Intelligence (AIˆ3 2017)*, pages 100–104, *AI*IA Series*, CEUR Workshop Proceedings, Volume 2012.
Acceptance Rate: ∼65%

Summary

In this paper we firstly describe the common problems of online discussions, as we have already seen in Section 2.1. The next problem we tackle is that the same arguments always have to be said again, because there is currently no common way of "importing" entities from previous discussions into newer ones. Therefore, we propose the distribution of discussion-entities from one *aggregate*, e.g. an online news provider, to another. By this, the arguments from a discussion on aggregate A can, at least in part, be reused in a discussion hosted on aggregate B, and the other way round. Doing so is important to keep the original author, if an entity is reused. Furthermore, we provide the idea of an update mechanism, which allows for arguments to be reused by other users, while the distributed aggregates still keep track of the history of the discussion-entities. This approach is further being investigated and published in Meter et al. 2018c.

More challenges when it comes to distributed discussion-entities is the nature of an argument, which depends on the way they are being collected and converted. For example, many arguments are *context-sensitive*, i.e. they are not universally usable, because they rely on other statements surrounding them. Therefore, importing non-universally usable entities can be a problem, leading to aggregates that have to define an import policy, which describes how to handle these cases.

The paper closes with an architecture sketch illustrating possible implementations of such a system concluding the idea of distributed arguments.

Personal Contribution

The initial idea of a decentralized network of arguments stems from Alexander Schneider. Developing and identifying the core challenges was jointly performed by both authors. Christian Meter mainly authored the challenges of updating distributed arguments as well as providing user-friendly interfaces. Alexander kept a focus on networking based challenges and the problem of context-dependent arguments. Also, he described the importance of an argument network.

Importance and Impact on this Thesis

With these ideas about distributed and reusable arguments we aim to change the way we are currently thinking about discussion-entities. Arguments are not just text strings, but valuable data, and should therefore be treated differently. During discussing this topic we came to the conclusion that reusing this data is a key feature to address multiple problems of current online discussions, specifically the presence of redundancy and fact-less arguments. As a side effect, the total amount of entities might be decreased, because parts of a discussion can simply be imported, and we will be able to compose conversations based on previous work.

This idea is of huge importance and the implementation of such a system is a necessary step to tackle current problems, which is why both of us kept working on the next steps in Meter et al. 2018c.

Reusable Statements in Dialog-Based Argumentation Systems

Alexander Schneider and Christian Meter

Heinrich Heine Universität Düsseldorf, Universitätsstr. 1, Düsseldorf 40235, Germany,
`firstname.lastname@hhu.de`,
WWW home page: `http://cn.hhu.de`

Abstract. Discussions on the Internet are usually conducted in isolation on a single platform, although there are many discussions on the same topic going on simultaneously all over the Internet. We argue that it is possible to connect similar discussions by reusing arguments, thus gaining a connected network of statements, supports and counterarguments which helps eradicate redundant and repetitive parts of common discussions. To achieve this goal we outline challenges that need to be solved and propose a possible architecture to tackle those challenges.

Keywords: dialog-based argumentation, arguments, statement reusability, argument networks

1 Introduction

Nowadays a lot of discussions are conducted online on social media, webpages of news outlets and forums. Those discussions are often unstructured and become hard to follow after they reach a certain size. Dialog-based argumentation systems like D-BAS [5] allow the user to formulate arguments while conducting a conversation with the system. A user can utilize any arguments that other participants of the discussion contributed to deliberate and express her opinion. As field tests of D-BAS have shown, more people participate when they can reuse arguments made by other participants compared to when they are required to formulate their own thoughts into a formal argument. The flaw with such dialog-based discussions is that they are localized and users thus can only re-use arguments made in their specific instance of the system. To solve this issue we propose an architecture to network several discussion and content providers, which host dialog based discussions. The goal is to generate the possibility of (automatic) argument exchange between those providers thus generating a network of reusable arguments and later on whole discussions. The thought of arguments as a persistent reusable resource which can be improved as time goes on is quite compelling. To achieve this goal it is imperative to design and implement this argument network in a fashion which does not appeal solely to argumentation experts, but rather to the general public and the content providers. Since such a system heavily relies on being widely distributed and being used by a lot of people that in turn create arguments, it is the foremost goal to design the system in a fashion which is suited for this target group.

This paper has a twofold purpose. The first is to argue for – and bring attention to – the importance and possibility of an interconnected argument network which can be widely used and distributed. The second is to raise awareness of the specific challenges arising when dealing with arguments which are distributed over several systems.

The remainder is structured as follows. We give an overview on related work in Sect. 2 followed by an outline on the importance of distributing and reusing arguments in Sect. 3. Following, we discuss open challenges for such a system in Sect. 4 and propose a possible architecture in Sect. 5 before concluding the paper in Sect. 6.

2 Related Work

There are a few papers about a system for storing and reusing arguments called "The Argument Web" [1, 2]. The main difference to our proposal is that the Argument Web aims mainly at storing discussions in databases for later uses by a multitude of tools, while we aim to actively distribute and propagate user-generated arguments to be used by other non-expert users in a dynamic network. Heras et al. [3] have researched the formalization of user-generated argumentation on social networks. While we also work with user generated arguments, we go the opposite way and require user-interfaces that facilitate the arguments to be input in an already formalized structure albeit being natural language as proposed by Meter et al. [6]. Similarly Toni and Torroni [4] researched a methodology to convert user-generated comments into arguments.

3 Importance of Distributed Arguments

Reuse of arguments in a dialog-based discussion could help the users deliberate more efficiently. The user can recycle arguments already made by others or be confronted with their opinion on a matter without the strain of necessarily formulating ones thoughts into a formal argument. Since discussions on the Internet are not carried out by experts in the field of argumentation, the quality of arguments varies considerably. Well written and structured arguments would probably be propagated more often and as such heighten the quality of future discussions. Furthermore, a lot of discussions on the same topic happen in parallel on the Internet. As an example, in 2016 there is a high number of discussions about the "Brexit" going on, since every news outlet published stories about it and most of them also allowed discussions on the articles of some sort. Factor in more private discussions on social networks, like Facebook, and the number grows even higher. All of those discussions contain numerous arguments and trains of thoughts that were already stated in another similar discussion somewhere else. If those discussions were at least partially linked, one probably would not see the necessity to restate the same opinions, but would just express their view by agreeing or disagreeing with the available statements or by reusing them in a new discussion. An as of yet untested but likely side-effect of this recycling could be that the users would reach a point where they can continue with a branch of the discussion which is "new" and produces original arguments and statements faster than without recycling.

Another advantage of an argument network would be that new discussions would not have to start empty, since they could be seeded by already ongoing arguments to similar discussions or whole parts of the same discussion at another argument provider. The content providers hosting the discussions would benefit as well, since arguments made on their platform and shared could contain a reference to the place of origin in turn incite traffic to the content providers and argument hosts.

4 Current Challenges

To distribute arguments, one faces unique challenges which are not encountered when dealing with arguments as a single entity belonging to one specific discussion. This section tries to describe the challenges that need solving to fully realized distributed arguments in a real world setting outside of academia.

Development of a Distributed Architecture. Naturally, for arguments to be distributed there has to be the technical foundation allowing content and argument providers to store arguments and subsequently share them. All possible architectures have to be performant enough to support a large number of providers sharing arguments simultaneously. We acknowledge that this challenge is more geared towards the networking community, but want to emphasize its importance nonetheless. We furthermore provide a sketch of a possible architecture in Sect. 5.

User-Friendliness. A system relying on the participants to reuse arguments has to provide the right tools making it as easy as possible for the participant. One example could be a kind of universal bookmarks. E.g. if a user participates in a discussion on news-outlet X and sees a clever argument that she likes, she should be able to mark it for future use during a discussion on any platforms Y and Z. Optimally this should be hardware independent so the user can fluently switch between devices. Another possible helper for reusing arguments could be a service which suggests existing arguments of other platforms while the user is typing. Although, this solution requires a knowledge of most arguments in the network, which could turn out as an impossible task to solve efficiently.

Update of Arguments. In a system where arguments propagate between different systems and hosts, there is also the problem of how to handle updated arguments. In a user-driven system arguments are subject to change because of spelling or grammatical errors. These changes should optimally propagate to all systems reusing said argument. If and how this happens depends mainly on the architecture. From a networking view the choices are to build a highly interconnected network where updates are distributed as widely as possible but require a structured network that needs to be maintained. The other end of the spectrum is a loosely related network of federated hosts that exchange updates at will. This solution has a low overhead but also does not necessarily distribute all updates. In our architecture sketch we use a federated network, which uses a subscription system for arguments and topics to receive updates. Furthermore, the community of an argument host can be allowed to curate the acceptance or rejection of propagated changes as the system is mainly user-driven.

Context-Dependence of Arguments. Ideally, we do not only want to reuse arguments but also automatically import all supports and attacks of a reused argument as this would deepen the discussion without any effort at all. The problem here is that some arguments posses a context, which makes it impossible to import more than the argument itself. For example in a discussion about raising the quality of life in a town with little money, there could be the argument A "Lets build a park, since it raises the quality of life". An attack B on this argument could be "A park is too expensive for the current town budget". Now there is a similar discussion going on in a more wealthy town. Some participant reuses A, because she finds it a compelling argument. If B is imported automatically as well, it does not fit because the context of the town in question having a tight budget does not apply. Possible solutions for this problem can be found with natural language processing techniques that try to determine whether statements posses context or are context-free. Another possible solution would be to allow the participant that imports the argument to choose whether attacks or supports shall be imported as well. Although this could have an adverse effect on the participation rates, since it heightens the amount of work for the participant.

5 Architecture Sketch

A possible architecture for a distributed argument network should consists of interchangeable parts or modules to accommodate the heterogeneous requirements of different content providers. The modules need to be exchangeable as long as they fulfill a certain set of requirements. The main modules we propose are the user interface, the execution logic engine, the database, and a module which we call the *aggregator*. The database is used for plain storage of arguments that a host collected over its lifetime. The database in turn connects to the aggregator, which has a multitude of tasks. The most important task of the aggregator is to communicate with the aggregators of other hosts to exchange arguments when needed and also tend to fetching and retrieving updates on existing arguments. For faster access the aggregator should also provide a cache of the most used arguments, to be able to quickly answer queries without the need to communicate with the database too often. Furthermore, the aggregator coordinates information flow between the user interface and the execution logic. When a user interacts with the system through the user interface, the provided data is forwarded to the aggregator which provides additional arguments if needed and queries the execution logic engine for the next steps before sending the result back to the user interface. As such the aggregator is the communication hub in the envisioned architecture. An explanation on how the execution logic engine works is out of scope for this paper, but can be found in detail in the D-BAS paper [5].

In general, the network that would form between discussion hosts would be a federated network, imitating the Web. A provider of content that is willing to host discussions can deploy an implementation of the proposed architecture. After that the different hosts start to connect loosely every time arguments are exchanged between them. The first exchanges are initiated through users recycling arguments they have seen on other hosts. This is the exact reason why the system needs to give a user the capability to "bookmark" arguments. Hosts that know each other can establish a more solid relationship by interchanging arguments based on set rules instead of on demand by users. Much as the web, a federation of every willing provider should be possible, regardless of the size or power of the provider. Whether a private web-blog or a huge media outlet or a social media network decides to provide an argument host should make no difference on the network and the users.

6 Conclusion

In this paper we argued for the need of a system that facilitates reuse of (user-generated) arguments and discussions. We emphasized the benefits of such a system and pointed out big challenges which need to be solved before putting such a system in place. We also provided the sketch of an architecture for such a system. The proposed architecture utilizes a federated network of content-providers which share user-generated arguments and discussions. For future research on this matter an enhanced prototype implementation of the proposed architecture incorporating as many solutions to the open challenges as possible offers itself up.

References

1. Bex, F., et al.: "Implementing the argument web." Communications of the ACM 56.10 (2013): 66–73.
2. Rahwan, I., et al.: "Laying the foundations for a world wide argument web." Artificial intelligence 171.10-15 (2007): 897–921.
3. Heras, S., et al.: "How Argumentation can Enhance Dialogues in Social Networks." COMMA. (2010): 267–274.
4. Toni, F., Torroni, P.: "Bottom-Up Argumentation." TAFA 7132 (2011): 249–262.
5. Krauthoff, T., et al.: "Dialog-Based Online Argumentation." COMMA. (2016).
6. Meter, C., et al.: "discuss: Embedding Dialog-Based Discussions into Websites." International Conference on Learning and Collaboration Technologies. Springer. (2017): 449–460.

5.2 EDEN: Extensible Discussion Entity Network

At last, we present the full paper about EDEN (Meter et al. 2018c):

Christian Meter, Alexander Schneider and Martin Mauve.

"EDEN: Extensible Discussion Entity Network"

In: *Proceedings of the 7th International Conference on Computational Models of Argument*, Volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 257–268, IOS Press.
Acceptance Rate: ∼50%

Summary

In the following paper we continue to pursue the ideas discussed in Section 5.1. At first, we describe the current situation concerning ephemeral arguments generated in forums or comment sections and criticize that solely academic projects exist, as a solution to this problem. Especially considering the fact that we need a collective change in the way of thinking, if we want to change the way online discussions are held.

The first change is that arguments need to be considered as valuable data, which should not be used longer than a discussion lasts. Secondly, collected statements are defined as immutable data structures containing an origin, the current text describing the statement, a deactivation flag (which can either be `true` or `false`) and a set of predecessors. By "changing" the content, a new instance of the statement is being produced and linked to its predecessor. Relations are similarly defined.

After the definition of our basic building blocks, we describe how these structures can be used to implement a *version control system*, similar to version control systems in software development. We are depicting a *fork*- and *update*-process, which comes into play when users copy a statement, modify the content, i.e. creating a new *version* of the statement, and then publish these changes. This new statement can now be "merged", i.e. the new version points to its predecessor and the original host of the statement is able to decide if the new statement should be the new official statement henceforward.

After discussing these thoughts, we present the reference implementation EDEN in Chapter 5 of the paper. EDEN can be used by multiple aggregate roots, e.g. online new publisher, and each instance of EDEN can join a network of EDEN instances to exchange the statements and relations, i.e. the arguments. To access these exchanged arguments, an instance of discuss can be used. Furthermore, D-BAS is used as the execution platform for the argumentation steps (our DGEP). This combines all available software tools to a complete stack, which is available for free and production ready.

Personal Contribution

The author of this thesis, Christian Meter, and his colleague Alexander Schneider both developed the architecture of EDEN in equal parts. Moreover, both of them implemented the software to make this project possible, whereby Christian focused on the messaging architecture that enables message passing between the single EDEN instances. Christian provided an overview on the idea of distributed arguments in this paper and focuses on the architecture and data structures of statements and links in this paper. Alexander described the distributed management of arguments as well as more basic concepts of EDEN. The remaining parts were jointly authored. Martin Mauve supervised parts of the paper and was also involved in the development process to rethink the way we should treat discussion entities.

Importance and Impact on this Thesis

EDEN provides an important part of the infrastructure which is necessary to reduce redundant arguments of discussions. Exchanging arguments is an essential part of achieving this goal. It changed our view on the whole problem and makes it crystal clear that we need this mechanism on our way to improve online discussions.

Therefore, EDEN is natively supported and used by discuss, so that argument interchange is being treated as a first class citizen in our discussion software stack. D-BAS does not need any changes, because EDEN is a transparent layer in our stack and discuss can directly connect to EDEN's API to access the statements from all connected providers. So, EDEN rounds up and completes our technology stack for dialog-based discussions.

September 2018

EDEN: Extensible Discussion Entity Network[1]

Christian METER, Alexander SCHNEIDER [2] and Martin MAUVE
Computer Networks Department
Heinrich-Heine University Düsseldorf, Germany
firstname.lastname@hhu.de

Abstract. Enabling the reuse of arguments as entities that can be shared across multiple Internet-based discussion platforms and that can be improved upon while they are being used and reused has many benefits ranging from easier participation in an online discussion to increasing the quality of arguments. In this paper we propose a mechanism that is able to support the large-scale reuse of arguments by providing distributed version control of argument data. Building on that mechanism we have designed and implemented *EDEN*, a framework which enables platform providers to easily network their discussions. EDEN is designed for real-world use and provides all tools necessary to enable the reuse of arguments and their interrelation for users and providers alike.

Keywords. massive online discussion, discussion networks, EDEN, discussion graphs

1. Introduction

Arguments and their interrelation are valuable resources. They require effort to craft and they reflect the knowledge and opinions of those that have contributed them. Furthermore, their value grows as a network of arguments and their interrelations increases in size. On the Internet this is currently not supported in an appropriate way. Most arguments are ephemeral postings in forums and comment sections of news media. Even dedicated argumentation websites do not allow for connecting arguments across multiple websites. In order to address this problem Bex, Lawrence, Snaith and Reed have introduced the notion of an *Argument Web* [6]. Unfortunately, the Argument Web has not (yet) gained sufficient traction and is limited to a set of research prototypes.

In this paper we argue that in order to have a larger impact, the Argument Web needs to be more than a way to specify, describe and reference arguments. In particular it has to take into account the specific needs of those that operate websites where argumentation is taking place and of the users that visit those web sites. As we shall discuss, this leads to a challenge on the system level that can be summarized in a simple question: How can arguments and their interrelations be managed as persistent resources in a distributed

[1] Submitted to IAT.
[2] Both first authors contributed in equal parts to this work.

(web-based) system? As an answer to this question we propose the idea of an *Extensible Discussion Entity Network* (EDEN).

EDEN is designed to provide persistent arguments and interrelations between them, which can be shared and reused, while incorporating the manifold requirements of users and platform providers. Those, sometimes contradicting, needs are usually not considered, when designing systems in the argumentation space. We believe that EDEN facilitates adoption for real-world scenarios.

The goals of this paper are twofold. First we would like to raise awareness for the fact that there are system-level challenges that need to be addressed in order to make the idea of an Argument Web work in a real-world setting. Second, we present a solution for the most important of those challenges, namely how to distribute and manage argument data in an heterogeneous Internet-based environment.

The remainder of the paper is structured as follows. Section 2 examines related work and compares it to our contribution. Next, we introduce our view on distributed argumentation and its stakeholders in Section 3. Section 4 then discusses a method for versioning arguments in a distributed environment. Following that, we present an implementation of EDEN which describes its functionality and specifics in Section 5. Finally, in Section 6 we conclude the paper with a summary and an outlook on future work.

2. Related Work

The idea of a connected network of arguments is not entirely new. The general idea for an "Argument Web" was established by Rahwan et al. [11] and further refined by Bex et al. [6]. Following the general idea a central database for the Argument Web was created by Lawrence et al. [8] which in turn interoperated with different applications belonging to the Argument Web [5,7]. The point where EDEN differs from that work is that we do not utilize a central database, which acts as a central interface for import and export for arguments in the AIF format. Instead we aim for dynamic exchange in a federated network of providers. Furthermore, EDEN is not bound to any special ontology, but instead focuses on arbitrary "atomic" entities.

There is also work by Rowe et al. [12] where the concept of reuse is anticipated by designing a system where it is possible to import and export arguments into and out of the Araucaria system on local instances.

Argument reuse has also been touched upon outside of the argumentation community. Kelly et al. [14] proposed the reuse of arguments via design patterns to ease the construction of safety cases and Smith and Harrison [13] proposed a system for reuse of descriptive arguments in hazard classification. To our knowledge EDEN is the first system to aim for reuse of arguments made by layman in a distributed online argumentation environment.

3. Reusable Arguments and their Environment

In order to be able to tackle the systems-level challenges posed by the idea of reusable arguments, we need a good understanding of the environment, where those arguments are created and (re)used. This environment consists of websites and web-based services that host discussions. In particular, this includes online newsmedia, social networks and discussion forums. We term these websites and web-based services *argument aggregators*, since they aggregate arguments provided by users in order to form online discussions.

Argument aggregators typically have policies on what an acceptable user-provided argument is and they have mechanisms in place that ensure that the contributions of the users adhere to those policies. The policies of different argument aggregators are quite heterogeneous, thus the same arguments might be acceptable for some content aggregators while others would consider them a violation of their policies. Furthermore, argument aggregators typically perceive the arguments provided by the users as a valuable commodity which helps gain page impressions and generate income, hence they are unlikely to be willing to share them, unless they get something in return like a reference to their web-site or something similar.

Arguments consist of statements that are linked to each other by different types of relations. They are regularly provided by the users of an argument aggregator. Arguments are also often linked to the content of the argument aggregator, e.g. they might pertain to a discussion regarding a blog entry or a news-media article. A specific argument is initially submitted by a single user to a single argument aggregator. However, any user might later on be willing to improve the argument, for example by correcting spelling errors or by making a statement more concise. The users might also want to use a given argument in another discussion, potentially hosted by a different argument aggregator.

Arguments are interconnected. Each argument, potentially, has numerous relations to other arguments. Furthermore, arguments might only be valid in a specific context. I.e., an argument might contain implicit information, that are not specifically stated. For example, the argument “Our labs are in bad shape, therefore we need to invest in new lab equipment.” includes implicit information about the condition of the author’s working environment since not all existing laboratories are in bad shape.

4. Distributed Management of Arguments

The characteristics of argument aggregators, users and arguments lead to challenges at the systems level that need to be addressed in order for the idea of persistent and reusable arguments to come true. The most prominent one is the development of a suitable architecture for the storage and distribution of arguments, where arguments are updated in an appropriate way, if they are used by multiple argument aggregators.

Since argument aggregators are independent entities that desire autonomous control over the arguments they store, show to their users and distribute to other argument aggregators, the architecture of a system for reusable and persistent arguments needs to be distributed. Given that arguments and their interrelations can be modified and improved upon over time, this immediately raises the question how their shared state can be managed.

One option is to take all proposed updates and calculate a resulting state that is then used by every argument aggregator. This, however, entails two problems. First, there needs to be a mechanism calculating a shared global state, which is a hard, but potentially solvable, problem in a distributed system. Second, all argument aggregators would have to agree unanimously on how to handle all updates – in particular whether to accept a given update or reject it. This is unlikely to be feasible in a real world environment.

If updates are optional, however, arguments may have different states at different aggregators. This inconsistent state is likely to cause problems. For example, an attack on an argument may be valid only for a certain variant of that argument that exists only on a subset of providers, since others modified it. Thus it is not clear how the attack can be reused in a distributed environment where aggregators have different versions of the attacked argument.

To solve this dilemma, we propose an approach derived from distributed source code versioning: arguments, or rather the statements and interrelations that make up the arguments, do have a version. An update produces a new version without modifying the original. The updated version refers to the original(s) as it's predecessor(s), effectively preserving history. This allows both for persistence, since no version is ever deleted and free choice of the content aggregators regarding what updates to accept.

In order to support distributed versioning of arguments, two problems have to be addressed. On the one hand, appropriate data structures are required that support the versioning of arguments. On the other hand there needs to be a mechanism to distribute information about new versions to those that might be interested in updates.

4.1. Data Structures for Versioning Arguments

In order to provide versioning for arguments we first determine the entities that make up a network of arguments. Those are statements and relations between statements. We then define an object to be a specific version of a specific entity. The data structures for storing objects have some common elements for both statement objects and relation objects: a global identifier $\mathcal{N}_{host}$ for the argument aggregator that created the current object (for example, the DNS host name of the argument aggregator), a local identifier $\mathcal{N}_{id}$, that uniquely identifies the entity stored in the object amongst all the entities that this argument aggregator has created objects for, and a version number $\mathcal{N}_{version}$ that indicates a specific version of the entity at this argument aggregator. Together those three values represent the object-id $\mathcal{N}$ which uniquely identifies a specific object. An important aspect of the object-id is that it can be determined locally and does not have to be coordinated amongst argument aggregators.

Furthermore, each object also has a flag d that indicates if it has been marked for deactivation. The latter is required since nothing should ever truly be deleted when doing versioning. Therefore a deletion of an object is just signaled by a specific version of that object where this flag is set. Providers can chose to follow the deactivation by making the object inaccessible to their users.

In addition to the information that is common to all objects, a statement object contains the following information:

- $\mathcal{P}$: a set of pointers to immediate predecessor versions, which is either a set of object-ids, or $\mathcal{P} = \emptyset$.
- $\mathcal{C}$: the data that makes up the statement, typically a plain text and meta information such as the author of the statement and the authors of modifications to the statement.

Summarizing, a statement-object can be fully described as a tuple $\langle \mathcal{P}, \mathcal{N}, \mathcal{C}, d \rangle$.

Relations include the following additional information:

$\mathscr{S}$: the relation's source, which is the object-id of any statement
$\mathscr{D}$: the destination of the relation, which is the object-id of any statement or another relation
t : the relation-type, e.g. "attack" or "premise-conclusion-relation"

A relation-object is thus described by: $\langle \mathscr{N}, \mathscr{S}, \mathscr{D}, t, d \rangle$. Relations are treated as immutable, they can only be created and deleted, but their content never changes. Therefore they do not need a predecessor.

We do believe that these data structures are sufficiently generic to capture arbitrary argumentation schemes, by utilizing $\mathscr{C}$ as a store for atomic entities of a scheme, and yet they provide all the information required to support versioning. They are also quite easy to extend if the need should arise.

4.2. Versioning Arguments in a Distributed Environment

An object (and thus a specific version of a specific entity) has an *authoritative argument aggregator*. This is the argument aggregator that created it and it can be easily determined by looking at $\mathscr{N}_{host}$ of that object. Another argument aggregator can import that object in order to integrate it into an argumentation that it hosts. After a provider imports an object, it can register with the authoritative aggregator for that object in order to receive updates regarding the entity contained in the object.

When an authoritative content aggregator updates an entity, it creates a new object for the new version of that entity with a new version number. It then notifies the argument aggregators that have registered with it regarding that entity. Those argument aggregators can then choose to accept the update or they can stick with the old version. This is a local decision that could be made by a dedicated moderator, the users of the argument aggregator or by means of a policy where one argument aggregator decides to trust another argument aggregator to provide reasonable updates.

If an entity is updated by an aggregator that is non-authoritative, a *fork* is created. A fork is a new object. For example if the original statement object was $St_1 = \langle \emptyset, id_x, \mathscr{C}, 0 \rangle$ with $id_x = \langle someaggregator.com, 42, 0 \rangle$, then the new fork-object including the new version of that entity could be $St_2 = \langle \{id_x\}, id_y, \mathscr{C}_2, 0 \rangle$ with $id_y = \langle anotheraggregator.org, 13, 0 \rangle$. The aggregator which created the fork is authoritative for that fork. When a fork of a statement is created, all relations belonging to the forked statement are copied and all instances of the forked statement are replaced by the fork in the copied statements. This does not update existing relations, but rather produce new ones specifically for the fork-object.

When an aggregator F creates a fork, it contacts the authoritative aggregator A of the object that was forked. A can decide to ignore the update. Then nothing happens and A remains authoritative for the original object while F is authoritative for the forked object. Or A can accept the update. In that case, it creates a new version of that entity by creating an appropriate object, which has an incremented version-number, updated content and the fork-object as its predecessor, to keep the version history accurate. As with all updates, the new object is then transmitted to all argument aggregators that have registered with the authoritative aggregator regarding that entity. In particular this is received by F. Once F realizes that its update has been accepted, it replaces the fork with the received update.

September 2018

4.3. Example of Fork and Update Processes

In order to illustrate how the proposed versioning scheme works, we now present an example showcasing the fork and update processes. The example begins as an aggregator with the global identifier *a.com* creates a statement which looks as follows: $S = \langle\emptyset, id_a = \langle a.com, 24, 0\rangle, \mathscr{C}_1, 0\rangle$. Now there are several cases that can occur.

4.3.1. Updating the Statement

Through a user-driven process, *a.com* decides to update the content of the statement S, producing new content $\mathscr{C}_2$. As a consequence an official updated statement-object $\langle\{id_a\}, \langle a.com, 24, 1\rangle, \mathscr{C}_2, 0\rangle$ is created and published to all other aggregators using S. Those aggregators decide individually whether they stick with the old version or update to the new one.

4.3.2. Creating a Fork

An aggregator *b.org* is using S and wants to update the statement's content to $\mathscr{C}_3$. A fork is now created which looks as follows: $\langle\{id_a\}, id_b = \langle b.org, 40, 0\rangle, \mathscr{C}_3, 0\rangle$. This fork is reported to the original aggregator *a.com*. In case *a.com* rejects the update, nothing more happens. If *a.com* accepts the update, it creates an updated version of S and sets the fork as a predecessor to preserve history – resulting in: $\langle\{id_b\}, \langle a.com, 24, 1\rangle, \mathscr{C}_3, 0\rangle$. This is then published to all other aggregators using S. Upon receiving the new object, *b.org* replaces the fork with the update, since its own changes have now been incorporated by *a.com*.

4.3.3. Simultaneous Forks and Updates

Continuing the example in Section 4.3.2, *c.net* is now also using S. It, too, has created an update to S with the content $\mathscr{C}_4$, which results in the object: $\langle\{id_a\}, id_c = \langle c.net, 1337, 0\rangle, \mathscr{C}_4, 0\rangle$. This fork is also communicated to *a.com*, which already updated S after accepting the fork from *b.org*. *a.com* can now choose to incorporate both forks in a new update where the content is then $\mathscr{C}_5$, thus producing $\langle\{id_b, id_c\}, \langle a.com, 24, 2\rangle, \mathscr{C}_5, 0\rangle$. In this version, both the objects from *b.org* and *c.net* are predecessors of the updated object. Figure 1 showcases the relations in this scenario. *a.com* could have also chosen to solely use the fork from *c.net* as the most current version 2, disregarding the changes of *b.org* included in version 1, effectively creating: $\langle\{id_c\}, \langle a.com, st_1, 2\rangle, \mathscr{C}_4, 0\rangle$. Again, this is published to all other aggregators using S. Upon receiving this, *b.org* and *c.net* decide whether they want to stick with their current version or update to the new one.

5. EDEN

This section introduces EDEN, the implementation of the aforementioned ideas. We briefly describe the basic concepts of EDEN before we lay out the modular architecture, several optimizations and first experiences of usage.

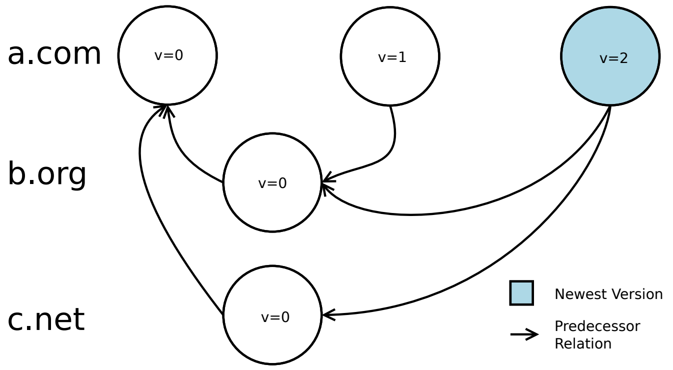


Figure 1. A visualisation of predecessor-relations between different forks and updates of a statement.

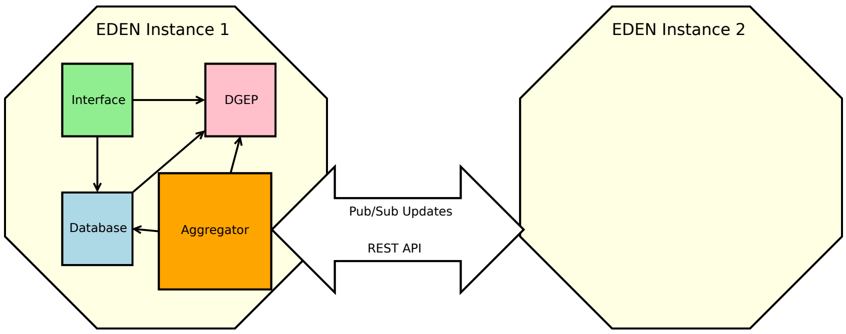


Figure 2. Dataflow between the modules of one EDEN instance. The dataflow with other EDEN instances is established via Pub/Sub and a REST API.

5.1. Basic Concepts

As described in the section above, EDEN is realized as a *federated network* of argument aggregators, where each aggregator is responsible for the state of its own data. Every argument aggregator that wants to enable its community to participate in the global argumentation network, can start up an EDEN instance, which discovers other instances through its initial whitelist and through foreign arguments discovered from those whitelisted instances. The most important task of EDEN is the management and exchange of local and foreign statements and relations. To this end the federated network maintained by EDEN has two logical layers – the local community of an aggregator and the global community spanning all available EDEN instances and their users.

Ideally, EDEN instances should be run by entities which are trusted by their users, like newspaper outlets, NGOs or other organizations. We do not, however, place any firm restrictions on which entities can run an EDEN instance.

We have developed EDEN with modularity in mind. EDEN therefore consists of independent modules, which can be exchanged, as long as they adhere to interface definitions between module "seams". Everything from the aggregator logic, the interface, the database to the execution logic can easily be customized and exchanged in individual EDEN deployments.

5.2. Architecture

The general architecture and dataflow of EDEN's architecture is shown in Figure 2. There are four main modules at work – each with its own purpose.

The interface module enables layman-users to participate in a discussion with their arguments. This in itself is a non-trivial challenge. We use *discuss*, described in Meter et al. [2], as an example implementation of the interface module. In order to allow users to easily import arguments, we present the user with similar arguments from the local and global community, while the user is trying to formulate their own thoughts into an argument by typing parts of it. Similarity here being the analogous and logical proximity of words being typed in respect to potential new arguments. There are many other ways how this support could be realized, e.g. by being able to bookmark arguments at one argument aggregator and then later on reuse these bookmarks in other argumentations at the same or a different argument aggregator. We chose this method to not impose any extra strain on the user in order to not deter them from using the system.

The aggregator module is, metaphorically speaking, the communication central and brain of the operation. All entities at one point pass through the aggregator module. Its duty can be divided into two sections. First, obtaining data from external EDEN instances and providing the local data back to them. Second, coordinating the internal flow of data to make sure it proceeds efficiently between the modules. Our implementation of the aggregator module provides a REST API to enable foreign EDEN instances to query it for data. We furthermore use the *RabbitMQ* publish/subscribe system for queues, to which the aggregators subscribe to be informed about updates to the subscribed entities.

The database module needs to store and efficiently provide heterogeneous data to the other modules. One could use traditional relational databases, but to simplify the storage and query of potentially big amounts of different data-types, EDEN uses an *Elasticsearch* database. One of the many advantages of Elasticsearch is the semantic search, which allows for sophisticated queries, e.g. searching for synonyms. This helps with the provisioning of relevant arguments in respect to the users input.

Finally there is a *Dialogue Game Execution Platform* (DGEP) as defined by Bex et. al. [4]. We use Krauthoff's *Dialog-Based Argumentation System* (D-BAS) [3] for this purpose. The DGEP is responsible for handling all necessary steps in a discussion, utilizing a predefined set of rules applying to a "natural" discourse. Through the modularity any DGEP could replace D-BAS inside the EDEN framework as long as it adheres to the interface conventions between the modules. Currently, the DGEP module also doubles as the module which creates structure data from user input. The choice for using D-BAS in the default version of EDEN is not made because of any architecture considerations, but because we simply needed to pick any one DGEP we could work with to provide a functioning implementation.

The communication with foreign EDEN instances is established in two different ways. If one instance is looking for an entity which may be stored at a different instance, it can query the remote aggregator via a REST API. This will provide it either with a "not-found" answer in case the entity could not be found or with the found entity and a publish/subscribe channel in the successful case. The querying instance can subscribe to the channel if desired to receive updates about new entities or changes in entities, i.e. new versions, thus making the pub/sub system responsible for push-based updates and the REST API for initial queries and pull-based updates.

September 2018

5.3. Statements and Links

EDEN uses the object types statement and relation[3] as described above. *Statements* are implemented as shown in Listing 1 with some required and some optional keys. The triplet of `[:aggregate-id, :entity-id, :version]` provides a unique address for a specific version of a statement entity. In particular this address can be used by non authoritative argument aggregators to refer to this version.

```
(s/def ::statement
  (s/keys :req [::author ::content ::created
                ::aggregate-id ::entity-id ::version]
          :opt [::ancestor-aggregate-id ::ancestor-entity-id
                ::ancestor-version]))
```

Listing 1: Definition of a statement.

Links are represented as immutable objects, which are defined by a type, source and destination in our implementation. The type represents the relation (e.g. attack, support, undermine, ...) and source and destination are references to objects in a specific version[4]. Since the links are immutable, they can be propagated alongside statements through the pub/sub channels and REST API. The aggregators can then resolve the link-references to the statements and show the users the appropriate versions[5].

```
(s/def ::link
  (s/keys
    :req [::author ::type ::created
          ::from-aggregate-id ::from-entity-id ::from-version
          ::to-aggregate-id ::to-entity-id ::to-version
          ::aggregate-id ::entity-id]))
```

Listing 2: Definition of a link.

5.4. Context Dependent Arguments

To properly import an argument into a foreign discussion, the reused data must be context-free. Our initial approach was to reuse statements and links in a way, that automatically included the reuse of all connected links and statements (e.g. attacks and supports) thus linking both argumentation graphs automatically. This does not always work, since statements may implicitly carry context pertaining to a specific discussion. For example if a family is discussing the acquisition of a pet the statement S_1: “Dogs are good family pets” may be used, with the corresponding attack A_1: “We do not have time to

[3] In the implementation relations were called links.

[4] The source is always a statement, while destination can be a link or statement.

[5] The current published version has the destination-version as an optional part for a link. This will change, according to the description in Section 4, in the next release.

walk a dog every day". The attack is true in the context of the family discussion, because it implicitly carries the information, that the family is too busy to care for a dog. If S_1 is now reused in the discussion of an animal-fan forum where the participants want to dedicate a lot of time to their pets and A_1 is automatically presented as an attack, it might not make a lot of sense.

There are different approaches which can be taken to solve this problem. The solution we choose to implement is an "intelligence of the masses" approach. This provides users with the ability to judge about context dependence of automatically imported statements in a review system, before they are fully added and presented to all other users in the discussion. The arguments can be judged one-by-one ordered in a queue accessible to the community members. This works as follows: When a user imports a statement, all other statements which have a relation with it are placed in this new queue. The reviewing users are presented with the statement at the head of the queue, which may be imported if its context-free, as well as with the statement that caused the import of the statement to be judged. The users can then vote to reject or to accept the import. Please note, that the users do not vote on their opinion regarding the content of a statement, but whether the import of it is sensible in the context of the discussion. If a majority of voters accept the import, the statement is fully added to the local discussion and its immediately related statements are placed in the queue. To not overflow the queue with a growing number of review cases, it is capped to a reasonable maximum number of review cases. If the queue is nearing its maximum, statements which are closest to manually imported ones are prioritized. This should prevent the case where one imported statement fills the queue solely with its related statements, while others are left out. The success of this procedure relies on the user's ability to make objective contributions regarding natural language arguments, which is a feasible assumption as shown in a field study [1] for the D-BAS system, where the users were quite capable in reviewing different aspects of reported statements and arguments. A similar approach to include the community is also heavily used on the StackExchange platforms, e.g. *StackOverflow*[6].

5.5. Further Optimizations

We also implemented some optimizations which help EDEN to better perform its tasks of fostering argument reuse.

We implemented a background entity crawler to optimize argument recommendations to the user. The crawler activates periodically when the instance has unallocated resources and queries foreign instances for yet unknown entities which are then indexed to enhance the lookup-time in the future. The crawler always tries to index the most relevant entities first. In our case this means e.g. statements which are directly – and if none can be found – indirectly related to already known statements. This is done because the chances are higher a user will import statements more closely related to statements already present in the discussion than otherwise. Random entities are queried when all related ones are already indexed.

The aggregator, furthermore, uses a tiered system for retrieval of entities to optimize the information-flow. If it is queried for an entity, the aggregator first attempts a lookup inside its cache. Upon failing to find the desired item in the cache, the lookup is directed to the database. If the entity can not be found in the local database either, it is retrieved

[6] `https://stackoverflow.com`

from a foreign EDEN instance. This guarantees that the entities are found as fast as possible, since slower queries to the database and to foreign instances are reduced. Of course the last tier of querying remote aggregators is omitted if the query originated from a foreign instance.

5.6. Hands-On Experience

EDEN was written entirely in Clojure and can be freely obtained at github.com[7]. It can be run without further installation from the *Docker* virtual environment, for which we provide the proper configuration. The Docker container also includes a D-BAS and a discuss instance, which are used as DGEP and interface of EDEN, as mentioned in previous Sections.

We conducted first small-scale tests between two and three instances in small mockup-environments running in different Docker containers. Each container was configured to simulate a physical instance on the same network and we used statements and links which were gathered in a field study using D-BAS [1] and split them up into different subsets used by distinct test-instances.

The tests were not meant as definitive performance simulations or a scientific study, but to get an inkling of how multiple EDEN instances behave together. As we expected, the exchange of arguments worked without any further complications and felt natural to the user. Overall the user-experience did not differ from a normal usage of discuss without the EDEN network – except for the larger selection of pre-formulated arguments – which is a positive sign that the user-facing parts are working as intended and do not inherently add any extra strain on the user. Naturally, this was only conducted to gather a general first experience and we will conduct further real-world tests in the future to obtain more scientifically robust data.

6. Conclusion and Future Work

In this paper we introduced EDEN as a framework to enable discussion-entity reuse between different argumentation platforms. We discussed the challenge of keeping a consistent state in a distributed environment and the resulting challenges for versioning arguments. Our work contains solutions for versioning arguments in a distributed network as well as a solution for context-dependence of entities. Furthermore, we introduced a working implementation of the EDEN framework which is open source and freely available to use. The implementation also contains several technical optimizations and performed successfully in first small-scale tests.

One main challenge that remains as future work is the deployment and evaluation of EDEN by real-world argument aggregators. We are currently in the process of negotiating with companies that provide software for online-participation processes such as participatory budgeting and urban planning. We do believe that this might be an excellent starting point for sharing arguments, since there are many distinct online-participation processes that share common topics. Real world adoption could also be furthered by adding DGEP modules for argument aggregation services like www.debatepedia.org or

[7] https://github.com/hhucn/eden

September 2018

www.procon.org. or by incorporating argument mining modules for unstructured natural language arguments from e.g. social media.

We also plan to release improved versions of EDEN. Improvements can be pursued by designing methods to ease the reuse of arguments for the users even further. A shared user-base between different EDEN instances could be pursued to facilitate adoption of the network. Additionally, the technical performance of the framework can be improved upon as well.

References

[1] Krauthoff, T., Meter, C., & Mauve, M. (2017). *Dialog-Based Online Argumentation: Findings from a Field Experiment.* Proceedings of the 1st Workshop on Advances In Argumentation In Artificial Intelligence co-located with XVI International Conference of the Italian Association for Artificial Intelligence. Bari, Italy.

[2] Meter, C., Krauthoff, T., & Mauve, M. (2017, July). *discuss: Embedding Dialog-Based Discussions into Websites*. In International Conference on Learning and Collaboration Technologies (pp. 449-460). Springer, Cham.

[3] Krauthoff, T., Meter, C., Betz, G., Baurmann, M. & Mauve, M. (2018, September). *D-BAS – A Dialog-Based Online Argumentation System*. Computational Models of Argument (pp. 325-336), Warsaw.

[4] Bex, F., Lawrence, J., & Reed, C. (2014, September). *Generalising argument dialogue with the Dialogue Game Execution Platform*. In COMMA (pp. 141-152).

[5] Bex, F., Snaith, M., Lawrence, J., & Reed, C. (2014). *Argublogging: An application for the argument web.* Web Semantics: Science, Services and Agents on the World Wide Web, 25, 9-15.

[6] Bex, F., Lawrence, J., Snaith, M., & Reed, C. (2013). *Implementing the argument web*. Communications of the ACM, 56(10), 66-73.

[7] Lawrence, J., Bex, F., Reed, C., & Snaith, M. (2012). *AIFdb: Infrastructure for the Argument Web.* In COMMA (pp. 515-516).

[8] Lawrence, J., Bex, F., & Reed, C. (2012). *Dialogues on the Argument Web: Mixed Initiative Argumentation with Arvina.* In COMMA (pp. 513-514).

[9] Rahwan, I., & Reed, C. (2009). *The argument interchange format.* In Argumentation in artificial intelligence (pp. 383-402). Springer, Boston, MA.

[10] Rahwan, I. (2008). *Mass argumentation and the semantic web.* Web Semantics: Science, Services and Agents on the World Wide Web, 6(1), 29-37.

[11] Rahwan, I., Zablith, F., & Reed, C. (2007). *Laying the foundations for a world wide argument web*. Artificial intelligence, 171(10-15), 897-921.

[12] Rowe, G., Macagno, F., Reed, C., & Walton, D. (2006). *Araucaria as a tool for diagramming arguments in teaching and studying philosophy*. Teaching Philosophy, 29(2), 111-124.

[13] Smith, S. P., & Harrison, M. D. (2002, April). *Improving hazard classification through the reuse of descriptive arguments*. In International Conference on Software Reuse (pp. 255-268). Springer, Berlin, Heidelberg.

[14] Kelly, T., & McDermid, J. (1998). *Safety case patterns-reusing successful arguments.*

Chapter 6

Conclusion

This section closes the dissertation and provides a brief overview on the results and findings of our research as well as some further directions for future work. The objective of this dissertation was to enhance the dialog-based online discussion approach DBOA and to introduce new ways for embeddings of dialog-based discussions into other non-standalone contexts in the Web. Moreover, during our research, we experienced that arguments are more than just textual statements in different combinations, but rather valuable data. Understanding this novel idea of arguments as a resource forms the last part of this dissertation.

During this thesis, six papers and one demo were accepted and published in peer-reviewed conference proceedings: Krauthoff et al. 2018, 2017; Meter et al. 2018a, 2017a, 2020, 2018c; Schneider and Meter 2017, 2019. Furthermore, the latest field-study is currently submitted to a conference.

Additionally, all software projects (D-BAS[1], discuss[2], EDEN[3], Jebediah[4], decide[5]), were published with an open-source license and are freely available. Furthermore, some projects are always online for live-testing the latest versions of our tools.

6.1 Results

During the first phase of this thesis, we researched how to improve the first draft of DBOA to make it usable for untrained users. So, we started to enhance the overall user experience of our argumentation system D-BAS and to refactor the code base in order to get a *stable argumentation core*, on which further applications may build on. After this, we conducted several field experiments with D-BAS and obtained a tight feedback loop with the participating users to achieve our initial goal: creating software for the *people*. With several changes in D-BAS and improvements as well as reductions in the way we conducted dialogs, we achieved a state in our software tools, which made us confident to think about and develop more applications for dialog-based discussions.

[1]Source code: `https://github.com/hhucn/dbas`, Demo: `https://dbas.cs.hhu.de`
[2]Source code: `https://github.com/hhucn/discuss`, Demo: `https://discuss.cs.hhu.de`
[3]Source code: `https://github.com/hhucn/eden`
[4]Source code: `https://github.com/hhucn/jebediah`
[5]Source code: `https://github.com/hhucn/decide`

This leads us to the second part of the dissertation, which deals with embeddings of software tools backed with DBOA, most importantly discuss. With discuss, we break through the limitations a standalone web application possesses and achieve more and better integration options than before. Integration into web contexts includes the possibility to *deeply interact with online articles*, i.e., using text references to react to specific parts of an article. These references enable novel bootstrapping options to jump right into a conversation without further knowledge about the discussion process. By this, we achieve, on the one hand, interaction options with the article itself and, on the other hand, concrete options to react to (parts of) an argument. Thus, we cover multiple types of interactions enabling *structured and fine-grained discussions in the Web*. Our findings have been tested with untrained participants in a concluding study described in Section 4.3, showing that the interaction options were easily understood and heavily used in the discussions. Compared to a common comment section tool, there still exists a usability-problem. Nevertheless, participants, who used discuss in a discussion, produced more than twice as many arguments compared to users using a conventional comment-section-systems. Therefore, our approach of dialog-based discussions produced first positive results, which need further work finalizing our academic projects in order to form marketable and production-ready products.

In the last part of this dissertation, we describe new thoughts on handling arguments. The idea of *arguments as valuable resources* changes the way arguments from different sources need to be treated. Current systems rarely enable re-using or referencing arguments from previous discussions, but this needs to change in order to reduce redundancy. Therefore, we present an entity-exchange system, to distribute a discussion's entities, i.e. statements and relations. This system is called EDEN, was implemented by us and is now freely accessible, including the source code. During further thinking about arguments as data, we realized that a new system to track ownership of arguments is necessary. Accordingly, we developed a *versioning system for arguments*, which allows adding new versions to previously discussed discussion entities. In doing so, we encourage reusing and enhancing arguments instead of writing a similar argument, standing for itself, without further references, but having no reference to other (previously discussed) arguments.

6.2 Future Work

Our research solves most questions formulated at the beginning of this dissertation by proposing new ideas, researching them and presenting academic prototypes. At this point, these prototypes performed quite well in several field-experiments, despite needing some improvements, especially concerning usability. In this section, we propose some questions that are still left open after this dissertation.

6.2.1 Enhancing Usability

Currently, our systems rely on our own vision of how a user interface should be developed. In our later work, we enhanced the interfaces relying on the feedback provided by the test-users. Nevertheless, we do not have the expertise to design meaningful interfaces, which are really focused on being used by humans. It is a necessary task to rework the interface with experts in the field of user experience. This is why we are currently looking for partners to help us with enhancing the interfaces of our projects. As mentioned earlier in Subsection 1.2.2 we acquired *Handelsblatt Research* as a partner aiming at enhancing the usability of our software and testing it in a real-world context.

6.2.2 Enhancing Overview of Positions

The first field experiment with D-BAS (Krauthoff et al. 2017) showed in particular that many positions are provided in controversial discussions, which causes difficulties in joining a discussion, due to a missing overview. Reducing the number of positions on the one hand or taking a closer look at the position in order to merge them on the other hand, remain as open questions we are currently working on. Our idea is to check for semantic similarity with the help of algorithms used in the area of machine learning. With these suggestions, we will be able to reduce the number of positions, due to the fact that users might choose an existing one if they are convinced that it is identical. A master's thesis, which is currently supervised by the author of this dissertation, might provide some insights and improvements.

6.2.3 Better Argument Proposals

During the process of discussing, the suggestion of already existing statements and positions might reduce the number of new arguments. This is already integrated into D-BAS and discuss, but we are currently only using either the *Levenshtein Distance* (Levenshtein 1966) or a not so well-trained semantic search engine. Nevertheless, the search engine falls back to Levenshtein or tracks synonyms of the words in the new argument, which provides only slightly better results.

Using the graph structure provided by our tools might enhance the suggestions of similar arguments and currently the author of this thesis is supervising a student research project targeting this approach.

The same problem persists with EDEN, which uses the same semantic search engine[6] we have configured for D-BAS.

[6] https://github.com/hhucn/dbas-search

6.2.4 Generating Summaries of (Sub-)Discussions

After a discussion, organizers should have a reasonable overview on the discussion's content. Currently, we are providing an argumentation map containing all statements, but we are missing further tools to work with this map. Therefore, we need approaches to extract and summarize the content of a discussion based on this graph-structure. A student research project and a master's thesis are currently supervised by the author of this dissertation to work on handy summaries constructed in real time. These summaries are being generated with simple techniques from the field of machine learning and still need further research to provide satisfactory results.

6.2.5 More (Production-Ready) Applications

Starting with a standalone website, a lightweight interface for web contexts and integrations into Social Networks is a good approach. Nevertheless, we need to further investigate possible usages in the real-world. Specific tools need to be developed using the same argumentation logic we have described in DBOA, in order to have a real impact on applications and discussions outside of academia.

Bibliography

Alsina, Victòria and José Luis Martí (2017). “The Birth of the CrowdLaw Movement: Tech-Based Citizen Participation, Legitimacy and the Quality of Lawmaking”. In: *Analyse und Kritik* 40.2, pp. 337–358. ISSN: 01715860. DOI: 10.1515/auk-2018-0019 (cit. on p. 1).

ARG-tech (Mar. 21, 2020). *OVA from ARG-tech*. URL: http://www.arg-tech.org/index.php/projects/ova-2/ (visited on 03/21/2020) (cit. on p. 13).

Bex, Floris, John Lawrence, and Chris Reed (2014a). “Generalising argument dialogue with the Dialogue Game Execution Platform”. In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press* 266, pp. 141–152. ISSN: 09226389. DOI: 10.3233/978-1-61499-436-7-141 (cit. on pp. 3, 6).

Bex, Floris, John Lawrence, Mark Snaith, and Chris Reed (2013). “Implementing the Argument Web”. In: *Communications of the ACM* 56.10, p. 66. ISSN: 00010782. DOI: 10.1145/2500891 (cit. on pp. 3, 15, 101).

Bex, Floris, Mark Snaith, John Lawrence, and Chris Reed (Mar. 2014b). “Argublogging: An application for the argument web”. In: *Journal of Web Semantics* 25, pp. 9–15. ISSN: 1570-8268. DOI: 10.1016/j.websem.2014.02.002. URL: http://www.sciencedirect.com/science/article/pii/S1570826814000079 (cit. on p. 14).

Budzynska, Katarzyna, Mathilde Janier, Juyeon Kang, Chris Reed, Patrick Saint-Dizier, Manfred Stede, and Olena Yaskorska (2014). “Towards Argument Mining from Dialogue”. In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press* 266, pp. 185–196. ISSN: 09226389. DOI: 10.3233/978-1-61499-436-7-185 (cit. on pp. 2, 18).

Bundesregierung via Facebook (Sept. 16, 2019). *Erderwärmung*. URL: https://www.facebook.com/Bundesregierung/posts/2463364120421976 (visited on 09/18/2019) (cit. on p. 1).

Cabrio, Elena and Serena Villata (2018). “Five years of argument mining: A Data-driven Analysis”. In: *IJCAI International Joint Conference on Artificial Intelligence* 2018-July, pp. 5427–5433. ISSN: 10450823. DOI: 10.24963/ijcai.2018/766 (cit. on p. 9).

Debate.org (Mar. 25, 2020). *The Premier Online Debate Website – Debate.org*. URL: https://www.debate.org/ (visited on 03/25/2020) (cit. on p. 15).

Debatepedia (Mar. 25, 2020). *Welcome to Debatepedia! – Debatepedia, the Wikipedia of Debates*. URL: http://www.debatepedia.org/ (visited on 03/25/2020) (cit. on p. 15).

DebateWise (Mar. 25, 2020). *DebateWise – Where Great Minds Differ*. URL: https://debatewise.org/ (visited on 03/25/2020) (cit. on p. 15).

DEMOS (Mar. 22, 2020). *DEMOS: Planning Solutions for smarter societies.* URL: https://demos-deutschland.de/ (visited on 03/22/2020) (cit. on p. 14).

Disqus, Inc. (Mar. 21, 2020). *Disqus – The number one way to build an audience on your website.* URL: https://disqus.com/ (visited on 03/21/2020) (cit. on p. 14).

Ebbinghaus, Björn (May 2019). "Decision Making with Argumentation Graphs". Master's Thesis. Heinrich-Heine-University Düsseldorf. DOI: 10.13140/RG.2.2.12515.09760/1 (cit. on pp. 11, 66, 141).

Ebbinghaus, Björn and Martin Mauve (2020). "decide: Supporting Participatory Budgeting with Online Argumentation". Submitted to the 8th International Conference on Computational Models of Argument (COMMA 2020) (cit. on p. 66).

Feger, Marc, Jan Steimann, and Christian Meter (Sept. 2020). "Structure or Content? Towards assessing Argument Relevance". In: *Proceedings of the 8th International Conference on Computational Models of Argument (COMMA 2020).* Perugia, Italy (cit. on p. 135).

Gordon, Thomas F and Douglas Walton (2006). "The Carneades argumentation framework–using presumptions and exceptions to model critical questions". In: *6th computational models of natural argument workshop (CMNA), European conference on artificial intelligence (ECAI), Italy.* Vol. 6. IOS Press, pp. 5–13 (cit. on p. 13).

GroupMap Technology Pty Ltd. (Mar. 22, 2020). *Pros and Cons List, Pros Cons List Template - GroupMap.* URL: https://www.groupmap.com/ (visited on 03/22/2020) (cit. on p. 13).

Guardian News & Media Limited (Apr. 12, 2016). *How we analysed 70m comments on the Guardian website.* URL: https://www.theguardian.com/technology/2016/apr/12/how-we-analysed-70m-comments-guardian-website (visited on 09/23/2019) (cit. on p. 1).

– (Aug. 7, 2019). *The only way to stop the catastrophe of a no-deal Brexit? Revoke article 50.* URL: https://www.theguardian.com/commentisfree/2019/aug/07/stop-catastrophe-no-deal-brexit-revoke-article-50 (visited on 09/23/2019) (cit. on p. 1).

idebate (Mar. 25, 2020). *International Debate Education Association (IDEA).* URL: http://idebate.org/ (visited on 03/25/2020) (cit. on p. 15).

Janier, Mathilde, John Lawrence, and Chris Reed (2014). "OVA+: An argument analysis interface". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press.* Vol. 266. 2014, pp. 463–464. DOI: 10.3233/978-1-61499-436-7-463 (cit. on p. 13).

Karacapilidis, Nikos and Dimitris Papadias (June 2001). "Computer supported argumentation and collaborative decision making: the HERMES system". In: *Information Systems* 26.4, pp. 259–277. DOI: 10.1016/S0306-4379(01)00020-5 (cit. on p. 14).

Kialo (Mar. 22, 2020). *Explore Popular Debates, Discussions and Critical Thinking – Kialo.* URL: https://www.kialo.com/ (visited on 03/22/2020) (cit. on p. 13).

Klein, Mark (2011). *How to Harvest Collective Wisdom on Complex Problems: An Introduction to the MIT Deliberatorium.* Technical Report: MIT Center for Collective Intelligence (cit. on p. 13).

Krauthoff, Tobias (2018). "Dialog-Based Online Argumentation". Dissertation. Heinrich-Heine-University Düsseldorf. URL: https://docserv.hhu.de/servlets/DocumentServlet?id=46330 (cit. on pp. 3, 17).

Krauthoff, Tobias, Michael Baurmann, Gregor Betz, and Martin Mauve (2016). "Dialog-Based Online Argumentation". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press.* Vol. 287. Potsdam, Germany, pp. 33–40. DOI: 10.3233/978-1-61499-686-6-33 (cit. on pp. 3, 6, 20, 21).

Krauthoff, Tobias, Christian Meter, Gregor Betz, Michael Baurmann, and Martin Mauve (Sept. 2018). "D-BAS – A Dialog-Based Online Argumentation System". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press.* Vol. 305. Warsaw, Poland, pp. 325–336. DOI: 10.3233/978-1-61499-906-5-325 (cit. on pp. 3, 6, 18, 20, 21, 123, 135).

Krauthoff, Tobias, Christian Meter, and Martin Mauve (2017). "Dialog-Based Online Argumentation: Findings from a Field Experiment". In: *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence.* Vol. 2012. CEUR-WS, pp. 85–99 (cit. on pp. 6, 18, 20, 21, 34, 123, 125, 135).

Kriplean, Travis, Jonathan Morgan, Deen Freelon, Alan Borning, and Lance Bennett (Feb. 2012). "Supporting reflective public thought with considerit". In: *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pp. 265–274. DOI: 10.1145/2145204.2145249 (cit. on p. 13).

Kriplean, Travis, Jonathan T Morgan, Deen Freelon, Alan Borning, and Lance Bennett (May 2011). "ConsiderIt: improving structured public deliberation". In: *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pp. 1831–1836. DOI: 10.1145/1979742.1979869 (cit. on p. 13).

Kunz, Werner and Horst W. J. Rittel (1970). *Issues as elements of information systems.* Vol. 131. Berkeley, California: Institute of Urban and Regional Development, University of California (cit. on p. 14).

Lackaff, Derek (2016). "Better Reykjavik - Open Municipal Policymaking". In: *Civic media: Technology, design, practice* May, pp. 229–234. DOI: 10.1111/jopp.12032 (cit. on p. 10).

Lawrence, John, Floris Bex, and Chris Reed (2012a). "Dialogues on the Argument Web: Mixed Initiative Argumentation with Arvina." In: *Frontiers in Artificial Intelligence and*

Applications, Computational Models of Argument, IOS Press. Vol. 245, pp. 513–514. DOI: 10.3233/978-1-61499-111-3-513 (cit. on p. 14).

Lawrence, John, Floris Bex, Chris Reed, and Mark Snaith (2012b). "AIFdb: Infrastructure for the argument web". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press* 245.1, pp. 515–516. ISSN: 09226389. DOI: 10.3233/978-1-61499-111-3-515 (cit. on pp. 15, 101).

Lawrence, John and Chris Reed (2020). "Argument Mining: A Survey". In: *Computational Linguistics* 45.4, pp. 765–818. DOI: 10.1162/coli_a_00364 (cit. on pp. 9, 12).

Levenshtein, Vladimir I (1966). "Binary codes capable of correcting deletions, insertions, and reversals". In: *Soviet physics doklady*. Vol. 10. 8, pp. 707–710 (cit. on p. 125).

Liebeck, Matthias, Katharina Esau, and Stefan Conrad (2016). "What to Do with an Airport? Mining Arguments in the German Online Participation Project Tempelhofer Feld". In: pp. 144–153. DOI: 10.18653/v1/w16-2817 (cit. on pp. 2, 10).

Lippi, Marco and Paolo Torroni (2016). "Argumentation mining: State of the art and emerging trends". In: *ACM Transactions on Internet Technology* 16.2. ISSN: 15576051. DOI: 10.1145/2850417 (cit. on pp. 2, 18).

Lührs, Rolf, Thomas Malsch, and Klaus Voss (2001). "Internet, discourses, and democracy". In: *Annual Conference of the Japanese Society for Artificial Intelligence, JSAI 2001: New Frontiers in Artificial Intelligence*. Springer, pp. 67–74. DOI: 10.1007/3-540-45548-5_9 (cit. on p. 14).

Macagno, Fabrizio, Chris Reed, and Douglas Walton (2007). "Argument Diagramming in Logic, Artificial Intelligence, and Law". In: *Knowledge Engineering Review* 22.1, pp. 87–109. URL: https://ssrn.com/abstract=1751655 (cit. on p. 11).

Medium Corporation (Mar. 22, 2020). *Medium – Get smarter about what matters to you*. URL: https://medium.com/ (visited on 03/22/2020) (cit. on p. 14).

Meter, Christian, Björn Ebbinghaus, and Martin Mauve (Sept. 2018a). "Jebediah – Arguing With a Social Bot". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument*. Vol. 305. Warsaw, Poland: IOS Press, pp. 467–468. DOI: 10.3233/978-1-61499-906-5-467 (cit. on pp. 6, 82, 123, 135).

Meter, Christian, Tobias Krauthoff, and Martin Mauve (2017a). "discuss: Embedding dialog-based Discussions into Websites". In: *Learning and Collaboration Technologies. Technology in Education. LCT 2017. Lecture Notes in Computer Science, vol 10296.* Springer, Cham. DOI: 10.1007/978-3-319-58515-4_34 (cit. on pp. 6, 68, 123, 135).

Meter, Christian, Tobias Krauthoff, and Alexander Schneider (Oct. 2018b). "Dialogbasierte Online-Diskussionen". In: Deutsche Verwaltungspraxis, DVP 11/18, vol. 69 (cit. on p. 136).

Meter, Christian, Alexander Schneider, Marc Feger, Jan Steimann, and Martin Mauve (2020). "discuss vs. Disqus: Evaluating Dialog-Based Discussions Against a Comment-Based System". Manuscript submitted for publication (cit. on pp. 2, 86, 123, 136).

Meter, Christian, Alexander Schneider, Philipp Hagemeister, and Martin Mauve (2017b). "Tor is not enough: Coercion in Remote Electronic Voting Systems". In: *arXiv preprint arXiv:1702.02816* (cit. on p. 136).

Meter, Christian, Alexander Schneider, and Martin Mauve (Sept. 2018c). "EDEN: Extensible Discussion Entity Network". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press.* Vol. 305. Warsaw, Poland, pp. 257–268. DOI: 10.3233/978-1-61499-906-5-257 (cit. on pp. 7, 102, 103, 109, 123, 135).

Mullick, Ankan, Sayan Ghosh, Ritam Dutt, Avijit Ghosh, and Abhijnan Chakraborty (2019). "Public sphere 2.0: Targeted commenting in online news media". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11438 LNCS, pp. 180–187. ISSN: 16113349. DOI: 10.1007/978-3-030-15719-7_23. arXiv: 1902.07946 (cit. on p. 5).

Neugebauer, Daniel (2018). "DABASCO: Generating AF, ADF, and ASPIC instances from Real-World Discussions". In: *Frontiers in Artificial Intelligence and Applications, Volume 305: Computational Models of Argument.* IOS Press. DOI: 10.3233/978-1-61499-906-5-469 (cit. on p. 51).

Pollock, John L. (1987). "Defeasible Reasoning". In: *Cognitive Science* 11.4, pp. 481–518. ISSN: 0364-0213. DOI: https://doi.org/10.1016/S0364-0213(87)80017-4 (cit. on p. 12).

Prakken, Henry and Gerard Vreeswijk (2001). "Logics for Defeasible Argumentation". In: *Handbook of Philosophical Logic*, pp. 219–318. DOI: 10.1007/978-94-017-0456-4_3 (cit. on p. 12).

Rahwan, Iyad, Fouad Zablith, and Chris Reed (2007). "Laying the foundations for a World Wide Argument Web". In: *Artificial Intelligence* 171.10-15, pp. 897–921. ISSN: 00043702. DOI: 10.1016/j.artint.2007.04.015. URL: http://linkinghub.elsevier.com/retrieve/pii/S0004370207000768 (cit. on pp. 3, 15, 101).

rbutr (Mar. 22, 2020). *Find rebuttals and promote critical thinking with rbutr.* URL: http://rbutr.com/ (visited on 03/22/2020) (cit. on p. 14).

Reddit Inc. (Mar. 22, 2020). *reddit: the front page of the internet.* URL: https://reddit.com/ (visited on 03/22/2020) (cit. on p. 13).

Reed, Chris (2006). "Preliminary Results from an Argument Corpus". In: *Linguistics in the Twenty-First Century.* Cambridge Scholars Press, in cooperation with Centro de Lingüística Aplicada de Santiago de Cuba, pp. 185–196. ISBN: 1904303862 (cit. on p. 15).

Reed, Chris and Glenn Rowe (2004). "Araucaria: Software for argument analysis, diagramming and representation". In: *International Journal on Artificial Intelligence Tools* 13.04, pp. 961–979. DOI: 10.1142/S0218213004001922 (cit. on p. 13).

Renton, Alastair and Ann Macintosh (2007). "Computer-supported argument maps as a policy memory". In: *The Information Society* 23.2, pp. 125–133. DOI: 10.1080/01972240701209300 (cit. on p. 11).

Sæbø, Øystein, Jeremy Rose, and Leif Skiftenes Flak (2008). "The shape of eParticipation: Characterizing an emerging research area". In: *Government Information Quarterly* 25.3, pp. 400–428. ISSN: 0740624X. DOI: 10.1016/j.giq.2007.04.007 (cit. on p. 9).

Sauf Pompiers Ltd. (Mar. 22, 2020). *MindMup: Argument Visualization*. URL: https://www.mindmup.com/ (visited on 03/22/2020) (cit. on p. 13).

Schneider, Alexander and Christian Meter (2017). "Reusable Statements in Dialog-Based Argumentation Systems". In: *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence*. Vol. 2012. CEUR-WS, pp. 100–104 (cit. on pp. 7, 102, 123, 135).

– (June 2019). "Various Efforts of Enhancing Real World Online Discussions". In: *Reason to Dissent: Proceedings of the 3rd European Conference on Argumentation (ECA 2019), Volume III*. Groningen, Netherlands: College Publications. ISBN: 978-1-84890-333-3 (cit. on pp. 7, 51, 123, 135).

Schneider, Alexander, Christian Meter, and Philipp Hagemeister (2017). "Survey on remote electronic voting". In: *arXiv preprint arXiv:1702.02798* (cit. on p. 136).

Schneider, David C, Christian Voigt, and Gregor Betz (2007). "Argunet-a software tool for collaborative argumentation analysis and research". In: *7th Workshop on Computational Models of Natural Argument (CMNA VII)* (cit. on p. 13).

Shum, Simon Buckingham (2003). "The Roots of Computer Supported Argument Visualization". In: *Visualizing Argumentation*. Springer, London, pp. 3–24. ISBN: 1852336641. DOI: 10.1007/978-1-4471-0037-9_1 (cit. on p. 11).

Statista Inc. (Jan. 2020). *Number of monthly active Facebook users worldwide as of 4th quarter 2019*. URL: https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/ (visited on 02/09/2020) (cit. on p. 6).

Steinbach, Malte, Nadja Wilker, and Sabrina Schöttle (2019). "E-participation on the local level–A census survey approach for researching its implementation". In: *Journal of Information Technology and Politics*, pp. 1–21. ISSN: 1933169X. DOI: 10.1080/19331681.2019.1676361 (cit. on p. 10).

Tan, Chenhao, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee (2016). "Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions". In: *25th International World Wide Web Conference, WWW 2016*, pp. 613–624. DOI: 10.1145/2872427.2883081. arXiv: 1602.01103 (cit. on p. 10).

Twardy, Charles (2004). "Argument maps improve critical thinking". In: *Teaching Philosophy* 27.2, pp. 95–116. ISSN: 21536619. DOI: 10.5840/teachphil200427213 (cit. on p. 11).

Wachsmuth, Henning, Martin Potthast, Khalid Al Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein (Sept. 2017). "Building an argument search engine for the web". In: *Proceedings of the 4th Workshop on Argument Mining*, pp. 49–59. DOI: 10.18653/v1/W17-5106 (cit. on p. 15).

Walton, D.N. (1996). *Argument Structure: A Pragmatic Theory*. Studies in Philosophy. University of Toronto Press. ISBN: 9780802071378. URL: https://books.google.de/books?id=lHmeQgAACAAJ (cit. on p. 11).

Wells, S. and C. A. Reed (2012). "A domain specific language for describing diverse systems of dialogue". In: *Journal of Applied Logic* 10.4, pp. 309–329. ISSN: 15708683. DOI: 10.1016/j.jal.2012.09.001 (cit. on p. 3).

Wells, Simon and Chris Reed (2007). "MAgtALO: Using agents, arguments, and the web to explore complex debates". In: *Agent, Web Services, and Ontologies Integrated Methodologies*, p. 76 (cit. on p. 14).

– (2008). "MAgtALO: an Agent-Based system for persuasive online interaction". In: *AISB 2008 Convention Communication, Interaction and Social Intelligence*. Vol. 1. Aberdeen, Scotland, p. 29 (cit. on p. 14).

Wyner, Adam, Katie Atkinson, and Trevor Bench-Capon (2011). "Towards a structured online consultation tool". In: *Electronic Participation*. Springer, Berlin, Heidelberg, pp. 286–297. ISBN: 978-3-642-23333-3. DOI: 10.1007/978-3-642-23333-3_25 (cit. on p. 14).

Personal Publications

Peer-Reviewed Conference Papers

Tobias Krauthoff, Christian Meter, and Martin Mauve (2017). "Dialog-Based Online Argumentation: Findings from a Field Experiment". In: *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence*. Vol. 2012. CEUR-WS, pp. 85–99

Christian Meter, Tobias Krauthoff, and Martin Mauve (2017a). "discuss: Embedding dialog-based Discussions into Websites". In: *Learning and Collaboration Technologies. Technology in Education. LCT 2017. Lecture Notes in Computer Science, vol 10296.* Springer, Cham. DOI: 10.1007/978-3-319-58515-4_34

Alexander Schneider and Christian Meter (2017). "Reusable Statements in Dialog-Based Argumentation Systems". In: *Proceedings of the 1st Workshop on Advances in Argumentation in Artificial Intelligence*. Vol. 2012. CEUR-WS, pp. 100–104

Tobias Krauthoff, Christian Meter, Gregor Betz, Michael Baurmann, and Martin Mauve (Sept. 2018). "D-BAS – A Dialog-Based Online Argumentation System". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press*. Vol. 305. Warsaw, Poland, pp. 325–336. DOI: 10.3233/978-1-61499-906-5-325

Christian Meter, Alexander Schneider, and Martin Mauve (Sept. 2018c). "EDEN: Extensible Discussion Entity Network". In: *Frontiers in Artificial Intelligence and Applications, Computational Models of Argument, IOS Press*. Vol. 305. Warsaw, Poland, pp. 257–268. DOI: 10.3233/978-1-61499-906-5-257

Alexander Schneider and Christian Meter (June 2019). "Various Efforts of Enhancing Real World Online Discussions". In: *Reason to Dissent: Proceedings of the 3rd European Conference on Argumentation (ECA 2019), Volume III*. Groningen, Netherlands: College Publications. ISBN: 978-1-84890-333-3

Marc Feger, Jan Steimann, and Christian Meter (Sept. 2020). "Structure or Content? Towards assessing Argument Relevance". In: *Proceedings of the 8th International Conference on Computational Models of Argument (COMMA 2020)*. Perugia, Italy

Peer-Reviewed Demo

Christian Meter, Björn Ebbinghaus, and Martin Mauve (Sept. 2018a). "Jebediah – Arguing With a Social Bot". In: *Frontiers in Artificial Intelligence and Applications, Computational*

Models of Argument. Vol. 305. Warsaw, Poland: IOS Press, pp. 467–468. DOI: 10.3233/978-1-61499-906-5-467

Manuscript Submitted for Publication

Christian Meter, Alexander Schneider, Marc Feger, Jan Steimann, and Martin Mauve (2020). “discuss vs. Disqus: Evaluating Dialog-Based Discussions Against a Comment-Based System”. Manuscript submitted for publication

Articles

Christian Meter, Tobias Krauthoff, and Alexander Schneider (Oct. 2018b). “Dialogbasierte Online-Diskussionen”. In: Deutsche Verwaltungspraxis, DVP 11/18, vol. 69

Technical Reports

Christian Meter, Alexander Schneider, Philipp Hagemeister, and Martin Mauve (2017b). “Tor is not enough: Coercion in Remote Electronic Voting Systems”. In: *arXiv preprint arXiv:1702.02816*

Alexander Schneider, Christian Meter, and Philipp Hagemeister (2017). “Survey on remote electronic voting”. In: *arXiv preprint arXiv:1702.02798*

Appendix

List of Figures

Glossary

API Application Programming Interface. 6, 21, 66–68, 82, 110

Argument Maps Argument maps are visual representations of argumentation structure extracted from free text. Usually they are represented as two-dimensional maps and should increase the understanding of a discussion. Find a more detailed description in Section 2.1. 13, 14

Arguments Arguments consist of a conclusion, a premise group and a relation. 12, 13, 18, 82, 83, 101, 109, 125, 142

Conclusions A statement, which is supported or attacked by a premise group, is called *conclusion*. 2, 12, 18, 82, 101, 141, 142

CSAV Computer-Supported Argument Visualization. 11

D-BAS Dialog-Based Argumentation System. 6, 7, 17, 18, 20, 21, 34, 35, 51, 66–69, 82, 86, 109, 110, 123, 125, 139, 141

DBOA Dialog-Based Online Argumentation. 3, 7, 17, 19–21, 34, 35, 51, 66, 68, 69, 82, 86, 87, 123, 124, 126, 141

decide *decide* in the version published in the master's thesis in Ebbinghaus 2019, is a web application, which uses the results from a discussion conducted in D-BAS to add decision options to it. Therefore, participants can vote for several positions previously discussed in D-BAS. 123

DGDL Dialogue Game Description Language. 3

DGEP Dialogue Game Execution Platform. 6, 17, 21, 68, 82, 109

discuss *discuss* is a lightweight embeddable interface to D-BAS, which enables dialog-based discussions in arbitrary web-contexts. 6, 7, 18, 68, 86, 101, 109, 110, 123–125, 139

Disqus This tool integrates a comment section into arbitrary websites via integration of a JavaScript file. Also, multi-page discussions are possible. 86

DMS Decentralized Moderation System. 21, 34, 35, 66

EDEN Extensible Discussion Entity Network. 7, 51, 86, 101, 109, 110, 123–125

IBIS Issue-Based Information System. 14

Jebediah *Jebediah* is a frontend assistant/social agent using D-BAS in its backend to enable DBOA in Social Networks. 6, 18, 82, 123, 139

Positions A statement which is used as a discussion opener, e.g. proposals for action, is called *position*. 11, 12, 17, 18, 35, 66, 67, 125, 139, 141

Premises or Premise Group If statements are used to support another statement, then they are called *premises*. One or more premises form a *premise group* if used together. 2, 12, 18, 35, 82, 101, 142

Relations and Inference Rules *Relations*, or *inference rules*, between conclusions and premises can either be supportive or attacking. 12, 18, 101, 109, 141

Social Agent A *social agent* allows interactions with users and provides information or interaction possibilities with other systems via an interface. Interactions can happen via text (then these programs are often called “chat bots”), or via other inputs, e.g. natural speech. 82, 83, 141

Statements They are (usually) user-generated and used in discussions. Statements receive their value by the contexts in which they are being used. 11, 12, 109, 125, 126, 139, 141, 142

Topics A set of arguments can be collected to a *topic*. This is usually done when people are discussing, but this is not a technical requirement. 11, 13, 17, 139

WoR Web of Reasons. 3, 20

Eidesstattliche Erklärung
laut §5 der Promotionsordnung vom 15.06.2018

Ich versichere an Eides Statt, dass die Dissertation von mir selbständig und ohne unzulässige fremde Hilfe unter Beachtung der „Grundsätze zur Sicherung guter wissenschaftlicher Praxis an der Heinrich-Heine-Universität Düsseldorf" erstellt worden ist.

Ort, Datum

Christian Meter