From Argument Search to Argumentative Dialogue: A Topic-independent Approach to Argument Acquisition for Dialogue Systems

Niklas Rach[1], Carolin Schindler[1], Isabel Feustel[1], Johannes Daxenberger[2], Wolfgang Minker[1], and Stefan Ultes[3]
[1]Institute of Communications Engineering, Ulm University, Germany
[2]Ubiquitous Knowledge Processing Lab, TU Darmstadt, Darmstadt, Germany
[3]Mercedes-Benz Research & Development, Sindelfingen, Germany
[1]{firstname.lastname}@uni-ulm.de
[2]daxenberger@ukp.informatik.tu-darmstadt.de
[3]stefan.ultes@daimler.com

Abstract

Despite the remarkable progress in the field of computational argumentation, dialogue systems concerned with argumentative tasks often rely on structured knowledge about arguments and their relations. Since the manual acquisition of these argument structures is highly time-consuming, the corresponding systems are inflexible regarding the topics they can discuss. To address this issue, we propose a combination of argumentative dialogue systems with argument search technology that enables a system to discuss any topic on which the search engine is able to find suitable arguments. Our approach utilizes supervised learning-based relation classification to map the retrieved arguments into a general tree structure for use in dialogue systems. We evaluate the approach with a state of the art search engine and a recently introduced dialogue model in an extensive user study with respect to the dialogue coherence. The results vary between the investigated topics (and hence depend on the quality of the underlying data) but are in some instances surprisingly close to the results achieved with a manually annotated argument structure.

1 Introduction

Argumentation is an interesting, yet challenging domain for dialogue systems. Existing systems address a multitude of tasks, including full scale debates against a human debater (Slonim et al., 2021), persuasion (Chalaguine and Hunter, 2020) and customer support (Galitsky, 2019). Due to the complex nature of this type of interaction, many systems rely on topic-specific argument structures that encode arguments and their relations to ensure a consistent and challenging interaction (Rach et al., 2018b; Sakai et al., 2020). Despite the advantages on the formal side, this dependency limits the range of topics that can be discussed by a system as the required structures are often either annotated by hand (Rach et al., 2019; Sakai et al., 2018b) or acquired in time-consuming data collections (Chalaguine and Hunter, 2019). This limitation renders the corresponding systems inflexible, especially in comparison to recent data-driven approaches in domains like question answering (Choi et al., 2018).

To address this issue, we propose a combination of argument search technology (Ajjour et al., 2019) with dialogue systems of the discussed kind. Our approach maps the list of pro and con arguments retrieved with an argument search engine for a given topic into a general tree structure that encodes bipolar relations (support and attack) between the individual arguments (see Figure 1). In doing so, our approach combines the strong points of both data-driven and formal models for argumentation and enables a corresponding system to discuss literally any topic on which the search engine can find suitable arguments. Throughout this work, we use the argument search engine ArgumenText (Stab et al., 2018) to retrieve pro and con arguments for a given topic from a large web crawl. In addition, we train and compare two classifiers to detect relations between pairs of the retrieved arguments which subsequently enables the aforementioned mapping into an argument structure.

The approach is evaluated with a formal model for persuasive dialogues that enables the generation of artificial discussions between two virtual agents. The resulting dialogues are then assessed in an extensive user survey with respect to their *coherence* and compared to the results achieved with an annotated structure. Although the annotated structure yields (as expected) an advantage over the automatically generated ones, the results are in some instances fairly close to each other. Besides, we observe varying results for the investigated topics, indicating a dependency of the approach on the available data.

Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 368–379

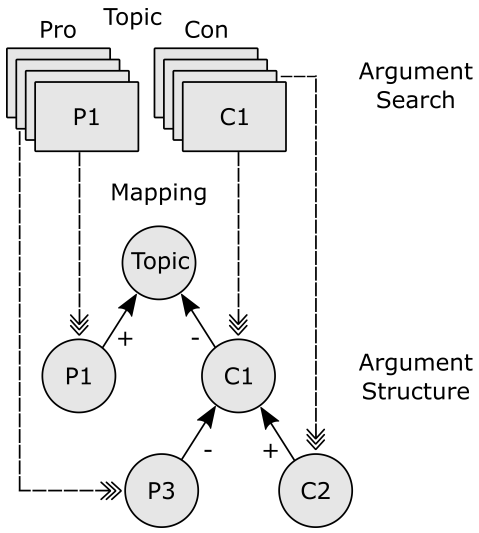

Figure 1: Mapping of argument search results to a tree structure with support (+) and attack (-) relations.

In summary, our contributions are:

- An approach to automatically generate argument structures from argument search results.
- An extensive evaluation of this approach in a challenging dialogue setup.

The remainder of this paper is as follows: Section 2 includes related work from the field of argumentative dialogue systems. The background on argument search and details about the utilized search engine are covered in Section 3, followed by a discussion of the proposed mapping in Section 4. Subsequently, we discuss the formal model for persuasion alongside the generation of artificial dialogues in Section 5 and the evaluation setup as well as the results in Section 6. The corresponding findings are discussed in Section 7.

2 Related Work

This section provides an overview of related work with a focus on argument retrieval for argumentative dialogue systems. The arguably most prominent system of this kind is the IBM project debater (Slonim et al., 2021), which is able to debate different topics with a human interlocutor. Although the system uses state of the art argument mining approaches to retrieve its arguments, it is tailored to the domain of debates and the utilized retrieval engine is currently not available to the public which hinders an application in other systems.

Approaches that rely on argument structures or graphs were investigated in different scenarios. The system by Rosenfeld and Kraus (2016) engages in a persuasive dialogue with human users. It utilizes a weighted bipolar argumentation framework with arguments collected in human discussions on the investigated topics. Chalaguine and Hunter (2019) conducted a crowdsourcing experiment to collect an argument graph for their desired domain and topic, whereas the argument structures employed in (Sakai et al., 2020) were generated using human annotators (Sakai et al., 2018b). Similarly, the systems of Rach et al. (2018b) and Aicher et al. (2021) also rely on annotated structures. In addition, human-generated argument graphs were considered by Hadoux and Hunter (2019), who selected arguments from multiple online sources manually for use in their system. Although the underlying formal frameworks of all these systems allow for complex dialogues, the topics that can be addressed are limited by the time-consuming generation of the argument structures. We propose an approach to generate structures of this kind automatically and independently of a specific topic.

In addition, data-based approaches were also investigated. The chatbot introduced by Rakshit et al. (2019) utilizes semantic similarity measures to retrieve arguments from an argument corpus to generate a response. A similar approach was compared to a generative model by Le et al. (2018) that was trained on a corpus of debate posts on various topics. Although especially the generative approach is focused on providing topic flexibility, aspects like user adaptation or strategy optimization as addressed in some of the previously discussed works are not (yet) considered in these systems. Our approach bridges the gap between formal and data-driven argumentation through a combination of argument search with formal models.

3 Argument Search

Argument search has recently evolved as an application from the field of argument mining (Lawrence and Reed, 2020). Argument search engines provide users with a (ranked) list of arguments related to a given search query, in some instances also including their stance/polarity towards the topic.

3.1 General Approach

Over the last years, different approaches to argument search were investigated that follow different paradigms (Ajjour et al., 2019). Systems introduced so far include the one developed in the scope

of IBM project debater (Levy et al., 2018), ArgumenText (Stab et al., 2018), args.me (Wachsmuth et al., 2017b), TARGER (Chernodub et al., 2019) and PerspectroScope (Chen et al., 2019). The general applicability of argument search engines in the context of dialogue systems was assessed in (Rach et al., 2020a) where ArgumenText and args.me were compared to a baseline system. Although a mapping into argument structures was not addressed, we use the discussed results to select a suitable search engine for the present work. Our model of choice is ArgumenText since it retrieves arguments on a sentence level (which is preferable in a dialogue context), performs reliable in comparison with the investigated baseline and additionally provides an API that allows for clustering the retrieved arguments thematically. In the following, a sentence retrieved by the search engine is denoted as argument ϕ and its polarity towards the topic as *stance*. An argument with a specified stance is denoted with *P* (pro) or *C* (con).

3.2 ArgumenText

ArgumenText provides multiple services for online argument mining that can be accessed via REST APIs[1]. We utilize the Search API, which retrieves arguments on a sentence level for a given search query. The engine utilizes a web crawl from the year 2016 based on CommonCrawl[2] to retrieve relevant documents and subsequently classify sentences in the documents as either pro, con or no argument (Stab et al., 2018). Besides the arguments and their stance, the search engine also provides multiple confidence values of which we use the one for stance (c_s) and argument detection (c_a) to derive the final confidence as $c = c_a \times c_s$ and rank the retrieved arguments accordingly.

In addition, we utilize ArgumenText's Cluster API to group the retrieved arguments thematically. It determines similarity scores for argument pairs which are then applied to form clusters based on aspects addressed within the arguments. The Cluster API relies on an optimized version of the Sentence-BERT method (Reimers and Gurevych, 2019) that makes use of an efficient bi-encoder that has been trained with additional samples ("Augmented SBERT") from a cross-encoder (Thakur et al., 2021). The utilized supervised approach to learn argument similarity was shown to outperform unsupervised approaches based on BERT embeddings by 10pp (Reimers et al., 2019).

4 From Arguments to Structures

In the following, the mapping of the retrieved arguments into an argument structure is discussed. Although some structures utilized by the systems discussed in Section 2 differ to a certain extent, they all require information about the relations between the individual arguments. We hence pursue a modular pipeline approach that first determines possible relations between the arguments and subsequently maps them into a specific structure. In case the required structure cannot be inferred from the herein discussed one, the second module can be adapted accordingly. This section builds on the work in (Schindler, 2020). The code of the complete pipeline is publicly available[3].

4.1 Target Structure

The herein considered target structure is based on the argument annotation scheme in (Stab and Gurevych, 2014), which distinguishes three different types of argument components (Major Claim, Claim, Premise) and two directed relations between them (support and attack). Each component has one unique relation towards another component but can be targeted by multiple others. To keep the structure as general as possible, we abstract from this framework in the sense that we are not distinguishing different component types for the retrieved arguments and only focus on finding the best fitting relation of each component towards another (or the main topic, i.e. the search query). Consequently, the resulting structure can be represented as a directed tree with the retrieved arguments as nodes, the relations as edges and the main topic as root (as depicted in Figure 1). To prevent isolated circles, we assume that each argument is (directly or indirectly) connected to the root.

4.2 Pipeline

Our pipeline takes arguments from an argument search engine (here ArgumenText) as input and outputs the above-discussed tree structure in an OWL file (Bechhofer, 2009). It first predicts relations between pairs of arguments and infers the final argument structure from them in a second step. We

[1] https://api.argumentsearch.com
[2] http://commoncrawl.org
[3] https://github.com/csacro/From-Argument-Search-to-Argumentative-Dialogue

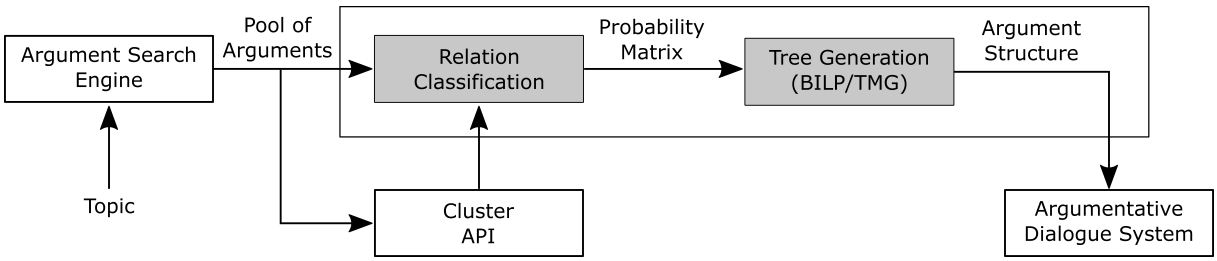

Figure 2: Sketch of the complete pipeline.

consider two configurations for the relation classification: The first predicts relations between all possible argument pairs and hence imposes no restrictions on the shape of the resulting tree. The second one utilizes the ArgumenText Cluster API (Thakur et al., 2021) to group the retrieved arguments prior to the relation classification. Consequently, only argument pairings within a cluster or with the main topic are considered during the relation classification and each cluster forms at least one new branch of the tree.

The relation classification is trained on a balanced subset of the corpus by Carstens and Toni (2015) as it labels sentence pairs with directed supportive, attacking or no relation. For the present task, the labels supportive and attacking are combined to a new label relation as the polarity can be inferred from the stance information provided by the search engine. We compare multiple classifiers including Support Vector Machine (SVM), Random Forest and Decision Trees on different feature sets with respect to their performance on the corpus. In addition, a BERT model (Devlin et al., 2019) is fine-tuned on the task. The detailed results are included in (Schindler, 2020) and we only include the best performing model as well as a strong baseline into the pipeline. The best performing classifier is the fine-tuned BERT model, reaching an average accuracy of 80.0% in a five-fold cross-validation. We select the SVM trained on BERT embeddings of argument pairs as the baseline due to its robust performance on a minimal feature set. The corresponding average accuracy in the five-fold cross-validation setup is 77.4%.

For the generation of the tree, we utilize the classifier confidence to compute probabilities for all estimated relations. Subsequently, we pursue two approaches to eliminate circles between arguments and derive the final tree structure: Binary Integer Linear Programming (BILP) optimizes the sum of the probabilities of the relations holding in the resulting tree under the structural constraints (Stab and Gurevych, 2017). In addition, we introduce Traversing and Modifying Graphs (TMG) which firstly identifies the most probable relation for every argument to another and connects them accordingly. Afterwards, it searches for circles as all resulting graphs which are not at least indirectly linked to the root contain exactly one such circle. In these circles, the node with the most probable relation to any node outside its graph is determined and the respective relation is redirected to this node. The complete pipeline is shown in Figure 2.

4.3 Preliminary Evaluation

To compare the above-selected approaches on the actual task, we conducted a preliminary annotation study. We retrieved 20 arguments from ArgumenText for the topics *nuclear energy is good* as well as *animal testing is good* and compared different combinations of the approaches to create the tree structure. Clustering prior to the relation classification was not considered in this step, as it is investigated thoroughly in the final evaluation. Five annotators without task-related background were asked to label each argument pair with a relation in the resulting tree structure in each of the annotation categories *contradiction*, *entailment*, *specificity*, *paraphrase* and *local relevance* with *yes* or *no*. The first four categories are based on an investigation of the interactions between semantic relations by Gold et al. (2019), the last category was proposed in (Wachsmuth et al., 2017a). As in this latter work, we use the labels of the three most agreeing annotators for each category in order to eliminate outliers.

The Fleiss' Kappa (Fleiss, 1971) values yields a substantial (0.66) up to perfect (0.82) agreement (Landis and Koch, 1977). A pair of arguments is concluded to actually hold a relation if it is rated with *yes* in at least one category by majority vote. For our baseline (SVM), this is the case with BILP

Speech Act	Attacks	Surrenders
claim(ϕ_i)	*why*(ϕ_i)	*concede*(ϕ_i)
why(ϕ_i)	*argue*($\phi_j \rightarrow \phi_i$), *argue_extend*($\phi_j \rightarrow \phi_i$)	*retract*(ϕ_i)
concede(ϕ_i)	-	-
retract(ϕ_i)	-	-
argue($\phi_j \rightarrow \phi_i$)	*why*(ϕ_j), *argue*($\phi_l \rightarrow \neg\phi_j$), *argue_extend*($\phi_l \rightarrow \neg\phi_j$)	*concede*(ϕ_j)
argue_extend($\phi_j \rightarrow \phi_i$)	*why*(ϕ_j), *argue*($\phi_l \rightarrow \neg\phi_j$), *argue_extend*($\phi_l \rightarrow \neg\phi_j$)	*concede*(ϕ_j)

Table 1: Communication language L_c of the utilized dialogue game for arguments of the investigated form.

as well as with TMG for 62.5% of the argument pairs. The BERT model correctly relates 75.0% of the argument pairs with TMG and 77.5% with BILP and we hence select the fine-tuned BERT model for the subsequent evaluation. It should be noted that BILP is highly time-consuming for large structures due to the underlying optimization problem. Since both approaches show similar performances, we only consider TMG in the final evaluation.

5 Argumentative Dialogue

To evaluate the complete pipeline in a dialogue setup, we generate artificial discussions between two virtual agents. The dialogues are created utilizing a recently introduced dialogue game for argumentation (Rach et al., 2020b) that extends the one introduced in (Prakken, 2005). It is chosen because it ensures a formally coherent selection of utterances, which means that all incoherent responses in the resulting dialogues can be clearly attributed to the retrieval pipeline. In addition, it offers the flexibility to go back to a previous utterance and respond with an alternative to the earlier response. This enables the agents to explore different branches of the tree structures and ensures a challenging setup for the evaluation.

5.1 Formal Framework

In the notation of (Prakken, 2005), the framework is formally described as $(\mathcal{L}, D)$, with $\mathcal{L}$ being a logic for defeasible argumentation that encodes the available arguments and their relations, i.e. the argument structure in the present case. The dialogue system proper D includes the communication language L_c and the protocol (rules) of the game. A game is played in turns and each turn consists of one or multiple game moves m_t. A temporally ordered sequence of moves is called a dialogue. Each move (except for the opening one) responds to one specific other move and either *attacks* or *surrenders* to this reference move. The communication language L_c includes the three attacking options *argue*, *argue_extend* and *why* as well as the two surrendering options *concede* and *retract*. The full communication language for arguments of the herein considered form, including the reply structure is shown in Table 1. For two arguments ϕ_i and ϕ_j, we therein denote a support relation with $\phi_j \rightarrow \phi_i$ and an attack relation with $\phi_j \rightarrow \neg\phi_i$.

To identify legal moves, the protocol determines whether the initial move is (logically) accepted or rejected in each dialogue based on a binary status (in/out). The current player can only respond to a move if an attacking reply to it affects the acceptability of the initial move. The turn of a player ends, if he or she successfully attacks an opponent move unless this attack includes an *argue_extend* move. The speech act type *argue_extend* allows players to anticipate *why* responses by introducing multiple supporting arguments in a single turn if they are available in the argument structure. A series of *argue(_extend)* moves is then called an *argument chain*.

5.2 Agent Strategy and Natural Language Generation

The agent strategies within the dialogue game and the natural language generation are adapted from Rach et al. (2019, 2020b), where we used similar setups to evaluate agent-agent dialogues. The strategy is based on probabilistic rules that prefer attacking replies over surrendering replies, attacking replies that address the immediate predecessor over delayed attacks and *argue(_extend)* over *why* moves. In addition, agents extend their attacks whenever possible, i.e. prefer *argue_extend* moves over *argue* moves. If multiple options with the same preference are available, the next move is selected randomly from this list. Due to its probabilistic nature, this strategy allows for the generation of different dialogues with a single argument structure which makes it a suitable choice for the present evaluation setup.

For the natural language generation, we use the sentences retrieved by ArgumenText as representation for the corresponding argument and select the formulation for the remaining moves randomly from a list of pre-defined templates. In case of a delayed response, the utterance also includes an explicit reference to the addressed one. As in the referenced work (Rach et al., 2020b), a series of *why* moves that responds to an argument chain is merged into a single utterance. An excerpt of a dialogue generated with an automatically retrieved structure on the topic *school uniforms are good* is shown in Appendix A.

6 Evaluation

This section discusses the evaluation of the artificial dialogues. We first introduce the study setup and discuss the results subsequently.

6.1 Setup

The first step in the evaluation is the selection of a meaningful set of evaluation categories. The ones utilized herein are based on the notion of dialogue coherence for conversational agents discussed by Venkatesh et al. (2017). The authors define a coherent response as one that is neither *irrelevant*, *incorrect* nor *inappropriate*. However, a direct application of these criteria is difficult in argumentative settings as for example the correctness of an argument is hard to assess. Therefore, each category is adapted into a yes/no question which directly evaluates utterance properties that are influenced by the retrieval pipeline. The resulting categories are as follows:

- Comprehensible: Do you understand what the speaker wants to say?
- Reference: Does the utterance address its reference?
- Polarity: Does the utterance contradict the speaker's position?

For the study, we implemented a web interface that presents the dialogues utterance-wise to the participants. In the beginning, participants received written instructions about the purpose of the survey and each of the above questions. In addition, a detailed example with manually generated arguments and explanations for the included ratings was provided to make the participants familiar with the setup. Each participant assessed three dialogues and was asked to rate the statement *The explanation/definition provided for the question was clear* for each evaluation category on a five-point Likert scale from 1 (totally disagree) to 5 (totally agree). In addition, participants were able to provide written feedback at the end of the survey.

We generated argument structures for seven different topics, namely *Nuclear Energy*, *Abortion*, *Self-driving Cars*, *School Uniforms*, *Death Penalty*, *Animal Testing* and *Marriage*. The first six topics are used to compare the two pipeline configurations (with and without clustering) and for a general assessment of the artificial dialogues. The topic *Marriage* on the other hand is used for a comparison to an annotated structure. The utilized reference structure includes 72 manually annotated arguments and relations between them from an idebate.org debate on the topic *Marriage is an outdated institution* (Rach et al., 2019). For each topic, we retrieved a pool of 60 arguments for the query *TOPIC is/are good* with ArgumenText and generated two structures per topic (with and without clustering). For each of the 14 automatically generated structures as well as the annotated one, we generated one reference dialogue for the evaluation and five additional dialogues. From the five additional dialogues, the one that has the least amount of arguments in common with the reference dialogue was added to the evaluation. Consequently, we arrived at a total of 30 dialogues that were divided into 10 groups of three dialogues each. To ensure similar conditions for all groups, the dialogues had a fixed length of 20 game moves. Participants were assigned to one of the 10 groups in order of appearance and we investigated seven raters per group, resulting in a total of 70 participants. The study was realized via clickworker[4] with participants from the UK (55) and the United States (15). The participants were aged between 18 and 67 years, 31 of them were female and 39 male.

6.2 Results

The study resulted in a total of 10,122 ratings over all 3 categories. We start the assessment of the results by computing the agreement over all three questions in each group with Fleiss' Kappa (Fleiss, 1971). The resulting agreement is rather low with a maximum of 0.46 (group 3) and a minimum of 0.14 (group 4), which indicates problems in the

[4] https://marketplace.clickworker.com (last accessed 12 March 2021)

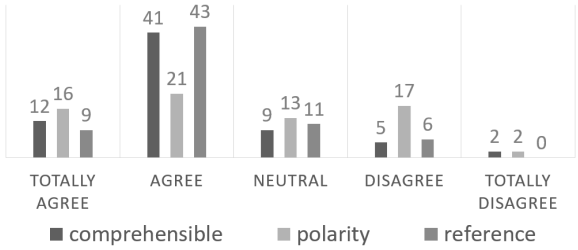

Figure 3: Responses on a five-point Likert scale from totally disagree (1) to totally agree (5) for all three evaluation questions and the statement *The explanation/definition provided for the question was clear*.

	G1	G2	G3	G4	G5
all	0.30	0.43	0.46	0.14	0.26
best 3	0.73	0.72	0.76	0.36	0.45
	G6	G7	G8	G9	G10
all	0.28	0.19	0.40	0.32	0.28
best 3	0.50	0.44	0.65	0.64	0.50

Table 2: Agreement derived with Fleiss' Kappa for all 10 groups (G1 - G10) and all annotators (all) as well as the three most agreeing annotators (best 3).

comprehensibility of the task. Consequently, we investigate the participants' self-report on the clarity of the task next. The corresponding results are shown in Figure 3. Although the majority of the ratings is either neutral or positive, there is also a certain percentage of negative ratings, especially for the *polarity* question. In total, 29 participants rated at least one category with *disagree* or *totally disagree*. Thus, we again consider the best agreeing three participants to derive the final score. The group-wise agreement for all and the best agreeing three participants is shown in Table 2. It can be seen that now all groups show a fair or better agreement (Landis and Koch, 1977). Given the subjective nature of the task (Wachsmuth et al., 2017a), we consider this a sufficient agreement for our evaluation and use the majority vote of the best agreeing three annotators in the following.

We proceed with a comparison of the two investigated pipeline configurations (with and without clustering) and subsequently compare the results of the automatically generated structures for the topic *Marriage* to the ones achieved with the annotated structure. We investigate each category/question separately and also compute the utterance-wise *coherence*. An utterance in the dialogue is fully coherent if it is comprehensible, addresses its reference and does not contradict the speaker's position, i.e. if it is rated with *yes*, *yes*, *no*. An example rating is included in Appendix A. For the comparison of the two pipeline configurations, we consider all topics with only automatically generated structures in the survey, namely *Nuclear Energy* (NE), *Abortion* (A), *Self-driving Cars* (SDC), *School Uniforms* (SU), *Death Penalty* (DP) and *Animal Testing* (AT). The corresponding ratio of positive and overall ratings is shown in Table 3.

It can be seen that the results are highly topic dependent, in direct comparison to each other and also in the effect of the clustering. The average over all topics (Overall) indicates a slight advantage of the group without clustering. However, a category-wise statistical comparison of the overall results with Fisher's exact test (Sprent, 2011) shows no significant difference between the two groups, indicating that (on average) both configurations perform equally well. Finally, the results of the annotated structure are compared to the results achieved with the automatically generated ones (with and without clustering) for the topic *Marriage*. We conduct a pairwise comparison of the three groups again with Fisher's exact test and a Benjamini-Hochberg correction (Benjamini and Hochberg, 1995) of the p-value. The corresponding results for all three structures are shown in Table 4. The annotated structure yields a perfect score of 1.00 for all categories, which is not surprising since it was tailored to dialogue setups. The comparison further indicates that the annotated structure outperforms the automatically generated ones, except for the *reference* category where no significant difference was found between the annotated structure and the automatically generated one without clustering.

7 Discussion

In the following, we discuss our findings from the previous section and the perspective of applications for the proposed method. As already mentioned, the results vary between the investigated topics for all evaluation categories. The difference between the individual topics can be attributed to the different sources the arguments are retrieved from and the resulting performance difference of the pipeline components. The effect of the clustering on the other hand is not so clear as both structures for a topic are based on the same pool of arguments. However, as the position of the arguments in the

	NE	A	SDC	SU	DP	AT	Overall
Comprehensible	0.89	0.92	0.94	0.94	0.66	0.88	0.87
Reference	0.80	0.97	0.91	0.78	0.91	0.85	0.87
Polarity	0.71	0.97	0.59	0.97	0.94	0.97	0.86
Coherence	0.60	0.87	0.47	0.75	0.53	0.76	0.67
Comprehensible	0.96	0.78	0.90	1.00	0.77	0.93	0.89
Reference	1.00	0.69	0.87	1.00	0.65	0.76	0.82
Polarity	0.82	0.78	0.87	0.48	0.94	1.00	0.82
Coherence	0.79	0.50	0.70	0.48	0.55	0.72	0.62

Table 3: Topic-wise results for the structures with (lower table) and without (upper table) clustering.

Results	annotated (a)	cluster (c)	no cluster (nc)
Comprehensible	1.00	0.68	0.83
Reference	1.00	0.68	0.86
Polarity	1.00	0.82	0.49
Coherence	1.00	0.43	0.34
p - values	a/c	a/nc	c/nc
Comprehensible	< 0.01	0.04	0.24
Reference	< 0.01	0.08	0.13
Polarity	0.01	< 0.01	0.01
Coherence	< 0.01	< 0.01	0.60

Table 4: Results for the annotated structure and the automatically generated ones on the topic *Marriage*. Upper table: Ratio of positive and overall ratings. Lower table: p-values of pairwise comparison with Fisher's exact test and Benjamini-Hochberg correction.

tree is directly influenced by the relation classification (and hence by the clustering as well), it varies between the structures with and without clustering. Therefore, the individual arguments can appear in a different context, which arguably also leads to a different perception through the study participants. On average, no significant difference between the two approaches could be found and the choice of the optimal configuration hence depends on the available data for each topic. The direct comparison with an annotated structure revealed room for improvement, especially with respect to the overall *coherence*. However, we also found that for the individual categories *comprehensible* and *reference*, the results achieved without clustering are fairly close to the performance of the annotated structure. Especially for the *reference* category, which is directly influenced by the herein introduced pipeline, the found difference between the annotated and the automatically generated structure without clustering was not statistically significant. In addition, the *coherence* results of the automatically generated structures on the topic *Marriage* were lower than for the other investigated topics, indicating that this was the most challenging topic for our approach. Although the above-discussed data dependency renders generalizations difficult, this *coherence* difference between the topic *Marriage* and the others indicates that the overall pipeline performance is closer to the one with annotated structures than suggested by the direct comparison.

As for the written feedback, multiple annotators reported confusing formulations of the argument as the major difficulty of the task. Since this is a direct consequence of the heterogeneous sources the arguments are retrieved from, it is hard to address in the pipeline. Therefore, approaches to automatically summarize or reformulate arguments (Bar-Haim et al., 2020; Schiller et al., 2021) could be beneficial to improve the performance.

Regarding applications, it can be seen that the proposed approach is quite flexible: Although a specific multi-agent setup was chosen for evaluation, the proposed pipeline itself has no dependency on this particular setting or the corresponding domain of persuasive dialogues. Therefore, it can be directly applied in other domains and scenarios as well if the respective dialogue system operates on structures of the retrieved kind. This includes for example systems in the opinion building domain (Aicher et al., 2021) or systems that combine argumentation with other types of dialogue like question answering (Sakai et al., 2018a). In addition, the proposed pipeline can be combined with methods that build on the investigated representation of arguments. In particular, the probabilistic rule-based strategy that was used in the evaluation setup can be extended or replaced with more sophisticated ones in compliance with the desired application. Examples in this regard are strategies optimized via reinforcement learning (Rach et al., 2018a) as well as argument selection based on semantics (Cayrol and Lagasquie-Schiex, 2005) or

user concerns (Chalaguine and Hunter, 2020). In light of the evaluation results, the main task for future work with respect to applications is hence the improvement of the pipeline performance to fully meet the quality requirements of the individual systems. However, as the proposed approach relies on argument search engines, it directly benefits from future developments in this area. Moreover, the addition of weights to arguments in the structure could further broaden the range of possible applications. The corresponding weights can for example be derived from the confidence scores of the pipeline components or through automatic approaches to assess argument quality (Wachsmuth et al., 2017a).

8 Conclusion

We have addressed the automatic generation of argument structures from argument search results for their use in dialogue systems. To this end, a pipeline was introduced that estimates relations between the retrieved arguments and maps them into a general tree structure. We explored two different configurations, namely with and without a prior clustering of the retrieved arguments and utilized a supervised learning-based relation classification to identify related argument pairs. For evaluation purpose, we generated 30 artificial dialogues over 7 different topics and assessed them in a crowdsourcing setup with respect to their *coherence*. The results indicate that the proposed pipeline depends on the quality of the available data but yields promising results for the majority of the investigated topics and at least one of the two investigated configurations (with and without clustering). In comparison to an annotated structure, we observed a similar performance for individual categories but also the expected room for improvement regarding the overall coherence. In summary, the proposed approach can be seen as a first step towards fully automatized argument acquisition for argumentative dialogue systems. Since it is based on argument search engines, it benefits directly from future improvements and developments in this area.

Future work will investigate automatic evaluation approaches that allow for an estimation of the pipeline performance given a specific topic. In addition, automatically generated structures will be applied in a dialogue system for an evaluation in direct interaction with human users.

Acknowledgments

Parts of this work have been funded by the Deutsche Forschungsgemeinschaft (DFG) within the project "How to Win Arguments - Empowering Virtual Agents to Improve their Persuasiveness", Grant Number 376696351, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

References

Annalena Aicher, Niklas Rach, Wolfgang Minker, and Stefan Ultes. 2021. Opinion building based on the argumentative dialogue system bea. In *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction: 10th International Workshop on Spoken Dialogue Systems*, pages 307–318. Springer Singapore.

Yamen Ajjour, Henning Wachsmuth, Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. 2019. Data acquisition for argument search: The args. me corpus. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, pages 48–59. Springer.

Roy Bar-Haim, Lilach Eden, Roni Friedman, Yoav Kantor, Dan Lahav, and Noam Slonim. 2020. From arguments to key points: Towards automatic argument summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4029–4039. Association for Computational Linguistics.

Sean Bechhofer. 2009. Owl: Web ontology language. In *Encyclopedia of Database Systems*, pages 2008–2009. Springer.

Yoav Benjamini and Yosef Hochberg. 1995. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300.

Lucas Carstens and Francesca Toni. 2015. Towards relation based argumentation mining. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 29–34, Denver, CO. Association for Computational Linguistics.

Claudette Cayrol and Marie-Christine Lagasquie-Schiex. 2005. On the acceptability of arguments in bipolar argumentation frameworks. In *European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 378–389. Springer.

Lisa A Chalaguine and Anthony Hunter. 2019. Knowledge acquisition and corpus for argumentation-based chatbots. In *CEUR Workshop Proceedings*, volume 2528, pages 1–14. CEUR Workshop Proceedings.

Lisa A Chalaguine and Anthony Hunter. 2020. A persuasive chatbot using a crowd-sourced argument graph and concerns. *Computational Models of Argument: Proceedings of COMMA 2020*, 326:9–20.

Sihao Chen, Daniel Khashabi, Chris Callison-Burch, and Dan Roth. 2019. PerspectroScope: A window to the world of diverse perspectives. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 129–134, Florence, Italy. Association for Computational Linguistics.

Artem Chernodub, Oleksiy Oliynyk, Philipp Heidenreich, Alexander Bondarenko, Matthias Hagen, Chris Biemann, and Alexander Panchenko. 2019. Targer: Neural argument mining at your fingertips. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 195–200. Association for Computational Linguistics.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Boris Galitsky. 2019. Enabling a bot with understanding argumentation and providing arguments. In *Developing Enterprise Chatbots*, pages 465–532. Springer.

Darina Gold, Venelin Kovatchev, and Torsten Zesch. 2019. Annotating and analyzing the interactions between meaning relations. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 26–36, Florence, Italy. Association for Computational Linguistics.

Emmanuel Hadoux and Anthony Hunter. 2019. Comfort or safety? gathering and using the concerns of a participant for better persuasion. *Argument & Computation*, 10(2):113–147.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Dieu Thu Le, Cam-Tu Nguyen, and Kim Anh Nguyen. 2018. Dave the debater: a retrieval-based and generative argumentative dialogue agent. In *Proceedings of the 5th Workshop on Argument Mining*, pages 121–130. Association for Computational Linguistics.

Ran Levy, Ben Bogin, Shai Gretz, Ranit Aharonov, and Noam Slonim. 2018. Towards an argumentative content search engine using weak supervision. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2066–2081. Association for Computational Linguistics.

Henry Prakken. 2005. Coherence and flexibility in dialogue games for argumentation. *Journal of logic and computation*, 15(6):1009–1040.

Niklas Rach, Saskia Langhammer, Wolfgang Minker, and Stefan Ultes. 2019. Utilizing argument mining techniques for argumentative dialogue systems. In *9th International Workshop on Spoken Dialogue System Technology*, pages 131–142. Springer.

Niklas Rach, Yuki Matsuda, Johannes Daxenberger, Stefan Ultes, Keiichi Yasumoto, and Wolfgang Minker. 2020a. Evaluation of argument search approaches in the context of argumentative dialogue systems. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 513–522. European Language Resources Association.

Niklas Rach, Wolfgang Minker, and Stefan Ultes. 2018a. Markov games for persuasive dialogue. In *Computational Models of Argument: Proceedings of COMMA 2018*, pages 213–220. IOS Press.

Niklas Rach, Wolfgang Minker, and Stefan Ultes. 2020b. Increasing the naturalness of an argumentative dialogue system through argument chains. *Computational Models of Argument: Proceedings of COMMA 2020*, 326:331–338.

Niklas Rach, Klaus Weber, Louisa Pragst, Elisabeth André, Wolfgang Minker, and Stefan Ultes. 2018b. Eva: A multimodal argumentative dialogue system. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pages 551–552. ACM.

Geetanjali Rakshit, Kevin K Bowden, Lena Reed, Amita Misra, and Marilyn Walker. 2019. Debbie, the debate bot of the future. In *Advanced Social Interaction with Agents*, pages 45–52. Springer.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *The 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)*, pages 3973–3983. Association for Computational Linguistics.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and clustering of arguments with contextualized word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 567–578. Association for Computational Linguistics.

Ariel Rosenfeld and Sarit Kraus. 2016. Strategical argumentative agent for human persuasion. In *ECAI*, volume 16, pages 320–329. IOS Press.

Kazuki Sakai, Ryuichiro Higashinaka, Yuichiro Yoshikawa, Hiroshi Ishiguro, and Junji Tomita. 2018a. Introduction method for argumentative dialogue using paired question-answering interchange about personality. In *Proceedings of the 19th annual SIGDIAL Meeting on discourse and dialogue*, pages 70–79. Association for Computational Linguistics.

Kazuki Sakai, Ryuichiro Higashinaka, Yuichiro Yoshikawa, Hiroshi Ishiguro, and Junji Tomita. 2020. Hierarchical argumentation structure for persuasive argumentative dialogue generation. *IEICE TRANSACTIONS on Information and Systems*, 103(2):424–434.

Kazuki Sakai, Akari Inago, Ryuichiro Higashinaka, Yuichiro Yoshikawa, Hiroshi Ishiguro, and Junji Tomita. 2018b. Creating large-scale argumentation structures for dialogue systems. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association.

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. 2021. Aspect-controlled neural argument generation. In *2021 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

Carolin Schindler. 2020. Argumentative relation classification for argumentative dialogue systems. Bachelor's thesis, Institute of Communications Engineering, Ulm University.

Noam Slonim, Yonatan Bilu, Carlos Alzate, Roy Bar-Haim, Ben Bogin, Francesca Bonin, Leshem Choshen, Edo Cohen-Karlik, Lena Dankin, Lilach Edelstein, et al. 2021. An autonomous debating system. *Nature*, 591(7850):379–384.

Peter Sprent. 2011. Fisher exact test. *International encyclopedia of statistical science*, pages 524–525.

Christian Stab, Johannes Daxenberger, Chris Stahlhut, Tristan Miller, Benjamin Schiller, Christopher Tauchmann, Steffen Eger, and Iryna Gurevych. 2018. ArgumenText: Searching for Arguments in Heterogeneous Sources. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 21–25. Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014. Annotating argument components and relations in persuasive essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021. Augmented SBERT: Data Augmentation Method for Improving Bi-Encoders for Pairwise Sentence Scoring Tasks. In *2021 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Anu Venkatesh, Chandra Khatri, Ashwin Ram, Fenfei Guo, Raefer Gabriel, Ashish Nagar, Rohit Prasad, Ming Cheng, Behnam Hedayatnia, Angeliki Metallinou, et al. 2017. On evaluating and comparing conversational agents. In *Advances in Neural Information Processing Systems, Conversational AI Workshop*.

Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Tim Alberdingk Thijm, Graeme Hirst, and Benno Stein. 2017a. Computational argumentation quality assessment in natural language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 176–187. Association for Computational Linguistics.

Henning Wachsmuth, Martin Potthast, Khalid Al Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein. 2017b. Building an argument search engine for the web. In *Proceedings of the 4th Workshop on Argument Mining (ArgMining 2017) at EMNLP*, pages 49–59. Association for Computational Linguistics.

A Examples

In the following, excerpts of an automatically retrieved argument structure, a dialogue generated with it and the corresponding majority ratings from the evaluation are included. The structure was retrieved with the complete pipeline (without clustering) and the search query *school uniforms are good*. Figure 4 shows the part used in the example dialogue, including three pro arguments (P1-P3), three con arguments (C1-C3) as well as their support (+) and attack (-) relations. The resulting dialogue excerpt and the respective majority ratings from the evaluation are shown in Table 5. When an argument is introduced, the corresponding source is indicated after the utterance in brackets together with the argument index that denotes its position in the argument structure. All sources are provided by ArgumenText as part of the search results and taken from the utilized web crawl.

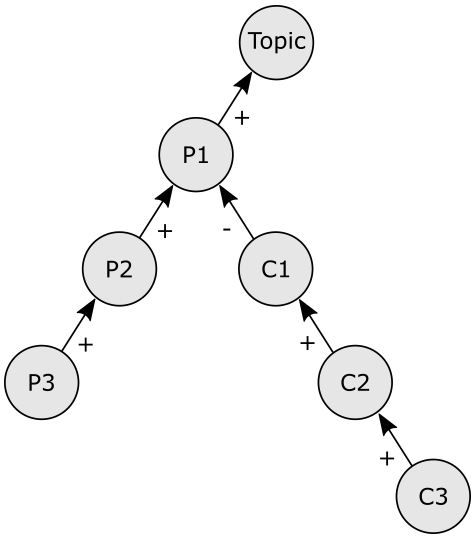

Figure 4: Excerpt of the argument structure used to generate the example dialogue in Table 5.

Speaker	Utterance	Majority Ratings	Coherent
A (pro)	*School uniforms are good.* (Topic)	-	-
A (pro)	*The uniform can help ease the problem about the fashion in school especially in a high school or a university that have a lot of teenagers together.* (P1, esl-lab.blogspot.com)	Comprehensible: *yes* Reference: *yes* Polarity: *no*	*yes*
A (pro)	*Wearing the same school uniform creates an image of unity and helps student feel accepted in the school.* (P2, ericmacknight.com)	Comprehensible: *yes* Reference: *yes* Polarity: *no*	*yes*
A (pro)	*It also shows unity and children feel included and it helps them to work as a team.* (P3, www.createdebate.com)	Comprehensible: *yes* Reference: *yes* Polarity: *no*	*yes*
B (con)	*You seem quite confident about that point. What makes you so sure?*	Comprehensible: *yes* Reference: *yes* Polarity: *no*	*yes*
B (con)	*In particular, there's one aspect of your argumentation that I have some doubts about. You said the uniform can help ease the problem about the fashion in school especially in a high school or a university that have a lot of teenagers together. Rude rules to wear school uniform is killing students' personality and it can be influenced for their future developing as bright personality.* (C1, www.createdebate.com)	Comprehensible: *yes* Reference: *yes* Polarity: *no*	*yes*
B (con)	*School Uniforms are not safe to wear.* (C2, www.createdebate.com)	Comprehensible: *yes* Reference: *no* Polarity: *no*	*no*
B (con)	*Uniform itself is also uncomfortable and during the winter it is very cold.* (C3, www.createdebate.com)	Comprehensible: *yes* Reference: *no* Polarity: *no*	*no*

Table 5: Artificial dialogue between the agents A and B generated with the dialogue game for argumentation and an automatically retrieved argument structure, including the majority answers to the questions *Do you understand what the speaker wants to say?* (Comprehensible), *Does the utterance address its reference?* (Reference) and *Does the utterance contradict the speaker's position?* (Polarity) as well as the corresponding coherence results.