Knowledge Acquisition and Corpus for Argumentation-Based Chatbots

Lisa Andreevna Chalaguine and Anthony Hunter

Department of Computer Science
University College London, London, UK
{ucablc3,a.hunter}@ucl.ac.uk

Abstract. Many of the conversations we have every day involve exchanges of arguments and counteraguments. In the context of artificial intelligence and argumentation theory, such phenomena fall into the area of dialogical argumentation. Conversational agents, also known as *chatbots*, are versatile tools that have the potential of being used in dialogical argumentation. We can assume that a chatbot would take a particular stance in the dialogue, opposing the stance of the user. In order to succeed, the chatbot also needs to be aware of various arguments and the interplay between them. Such knowledge can be represented by a directed graph, where nodes stand for arguments and arcs symbolise conflicts between them. The chatbot must be aware of both sides of the discussion, i.e. the arguments that it can play as well as ones that the user might have, to be able to formulate convincing responses. The availability of large argument graphs for research, however, is very limited. This means that researchers do not have corpora available which hinders the development of new chatbots and limits the effectiveness of existing ones. In this paper, we propose a method to acquire a large number of arguments in a graph structure using crowd sourcing. We evaluate this method in a study with participants and present a corpus which can be used for further research in computational argumentation and chatbot technologies for argumentation.

Keywords: argument acquisition · computational argumentation · automated chatbot knowledge acquisition · argument graphs · argument corpus

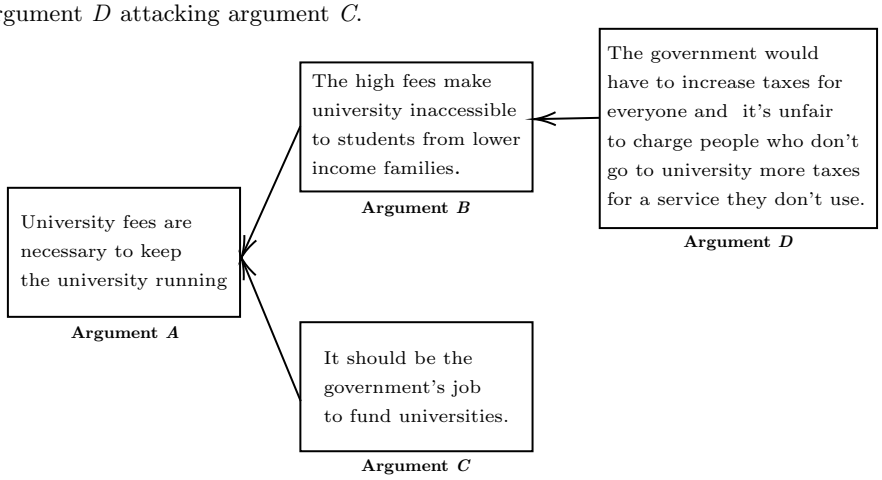
1 Introduction

The purpose of argumentation is to exchange different viewpoints or opinions, handle conflicting information and make informed decisions. The importance of argumentation has lead to the development of computational models of argument that aim to formalise aspects of argumentation within software. A key role for argumentation is in persuasion, and computational persuasion incorporates computational models of argument in software agents that can persuade people. This can be potentially valuable in roles such as behaviour change where the aim is to get the persuadee to make specific changes to the lifestyle (e.g. to eat

more fruit, to take more exercise, to commute by cycle, etc.) that can benefit them or those around them [1].

This calls for the development of methods for acquiring appropriate arguments and counterarguments that can be used as the chatbot's knowledge base. A situation involving argumentation can be represented by a directed graph, as proposed by Dung [2]. Each node represents an argument, and each arc denotes an attack by one argument on another. Such a graph can then be analysed to determine which arguments are acceptable according to some general criteria [3,4]. Figure 1 shows such an argument graph and the attack relationships between the arguments.

Fig. 1. Simple argument graph with arguments B and C attacking argument A and argument D attacking argument C.



Argument graphs are extensively studied in the computational argumentation literature, their acquisition, however, tends to be neglected. In order to have good quality dialogues, it is important that the argument graph has sufficient depth and breadth of coverage of the topic, so that the dialogue can proceed with more than one or two exchanges of argument per participant [5].

In order to constructs graphs using *real* arguments as opposed to made-up examples, arguments have to be acquired from real-life sources. This introduces the problem of where to obtain the relevant arguments for the argument graph. This highly depends on the topic and domain in question. In the behaviour change domain, for example, arguments on why eating a lot of fruits and vegetables is healthy, may be easily found in the professional healthcare literature. Arguments on why people do not follow a healthy diet, however, have to be obtained from people directly. In politics, arguments on why a new airport is necessary, will be advertised by the government, but again, counterarguments will have to be acquired from the people who oppose that project. On other topics, arguments

of both sides may be available in either the literature or the internet. Nevertheless, these arguments have to be extracted either manually or by the means of *argument mining* and somehow organised into an argument graph.

The creation of an argument graph for a chatbot knowledge base used for dialogical argumentation raises further issues, like (1) how to capture the majority of possible arguments without making the graph too big (in order to reduce search time to make the graph usable for a chatbot which has to reply fast to avoid irritating the user), (2) which arguments to include in the knowledge base and how to justify the inclusion of some and exclusion of others (e.g. noise and repetition of arguments), and (3) how to establish relations between arguments (the arcs of the graph). In order to address these questions, a corpus is needed which can be used for experiments.

Using forums for online discussions as source for chatbot knowledge base generation (for the rest of the paper we will assume that the chatbot will be used for dialogical argumentation) sounds tempting due to the large repositories which contain a great deal of human knowledge on many topics. However, using threads from websites like *reddit* for a chatbot knowledge base raises several problems. Firstly, unless it is a very popular topic it can take months to acquire a substantial number of arguments and risk not collecting any at all. Secondly, not all posts contain arguments. Often people share stories, ask or answer questions or make opinionated statements. Thirdly, long posts most likely contain several arguments and individual arguments would therefore have to be extracted with argument mining techniques. Lastly, the resulting graph is most likely to be very imbalanced. [6] graphically shows one of the largest reddit threads which contained over 33k comments. One can see that several branches continue for quite some time before branching out further into subbranches and some of the subbranches "die" rather quickly. This kind of structure is forced by the nature of the forum exchanges and the temporal and popularity aspects of the discussion. The resulting graph may therefore be rather deep but may not have sufficient breadth, thus still requiring extension from other sources.

1.1 Existing Approaches

Most chatbots are implemented using templates: for a specific question the chatbot provides an answer from a list of possible answers. These are usually hand coded and the construction of chatbot knowledge bases are therefore time consuming and difficult to adapt to new domains. There is limited research on fully automated chatbot knowledge acquisition. The most relevant to our research was proposed in [7]. It describes a method of using online discussion forums to extract chatbot knowledge, by automatically extracting the titles of threads and their replies, creating <thread-title, reply> pairs. In this way a knowledge base for a chatbot is constructed. These pairs, however, are not connected in a graph like structure and the chatbot's purpose was to answer questions and not engage in an argumentative dialogue. Chatbots that do make use of argumentation, usually assume an existing knowledge base where the counterarguments can be drawn from, or require researching the arguments and manually construct the

knowledge base. Climebot [8] (a conversational agent able to explain issues related to global warming), for example, relies on textual entailment to identify the best answer for a statement given by a human agent. The argumentative corpus from which the chatbot could choose from was extracted from three debating sites.

In our previous work [9] the arguments that the chatbot used were crowd sourced. The chatbot, however, was not aware of the users' counterarguments and was therefore not able to counter them, but only to present a new one which was not an attack to the user's argument. Hence, the chatbot was only able to acquire argument-counterargument pairs. The resulting argument graph would have extensive breadth but not go beyond two levels: the chatbot's arguments and the user's counterarguments.

A lot of research has been conducted on how to acquire arguments from the web and is generally referred to as *argument mining*. Argument mining exploits existing, and develops new, techniques from Machine Learning (ML) and Natural Language Processing (NLP); re-purposing and extending them to identify argument structures within text [10]. For an extensive overview on the latest research please refer to [11, 12]. Online generated discourse in forums or specific debating websites (e.g. *createdebate*[1] or *reddit*[2]) has also attracted research on argument mining [13, 14]. Threads from *reddit*, for example, have been used to create argument graphs for highlighting only the relevant arguments involved in a discussion [15] and assessment of persuasiveness [16]. IBM's *Debater* project [17] heavily relies on argument mining techniques and mine the arguments from published sources like Wikipedia. However, using forums for online discussions or published sources like Wikipedia as chatbot knowledge base for dialogical argumentation has its limitations, as already outlined above.

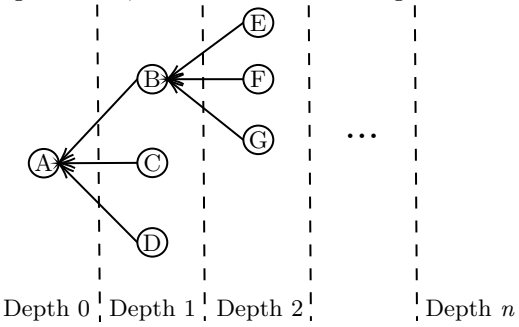
A more recent example of a chatbot that engages in dialogical argumentation is presented in [18] where the chatbot tries to persuade the user to cycle more. The chatbot's knowledge base was stored as an argument graph. The researchers undertook a web search on the pros and cons of city cycling and manually identified a number of arguments and attacks between them, which they encoded into an argument graph. Another example by the same researchers is presented in [19] on the topic of university fees in the UK which also involved a hand-crafted argument graph.

Another approach on how to collect arguments and construct an argument graph, without the use of online discussion forums or extensive research, was conducted using Dialog-Based Online Argumentation (D-BAS) and is described in [20]. Their resulting graph contains 265 arguments. It should be noted, however, that the researchers instructed the participants on how to counter previous arguments in order to obtain high-quality arguments and counterarguments. They also did not allow the repetition of arguments and motivated the participants to flag repetitions, as well as statements that should be revised, were off-topic or irrelevant, or abusive.

[1] http://www.createdebate.com/

[2] http://www.reddit.com/

Fig. 2. Representation of depths and attack relationships between arguments in our argument graph. Arguments *B, C* and *D* are counterarguments to *A*.



1.2 Proposed Solution

Our aim was to generate a corpus of arguments in a graph-like structure which we could use as a chatbot knowledge base in our further research where the chatbot would engage in an argumentation dialogue with real participants. In this paper, we propose a method to acquire a large number of arguments in a graph structure using crowd sourcing and present a corpus which can be used for further research in the computational argumentation domain. Apart from a minimum and maximum length, participants had no constraints when submitting arguments in order to create a big graph of natural language arguments.

In the rest of the paper, we describe our method to create an argument corpus on the topic of university fees in the UK and evaluate the quality of the obtained arguments in an experiment with crowd sourced participants.

2 Method

The depth of a graph is defined as the maximum number of arcs one can follow starting from the root. We decided to create a graph of depth 5, the root argument being depth 0. Starting from the root and following any path one will end up with a maximum of 5 arguments (excluding the root argument). The arguments in depth 1 attack the root argument and are therefore *against* keeping the university fees, the arguments in depth 2 attack the arguments in depth 1 and are therefore *for* keeping the fees and so on. Figure 2 shows a schematic representation of depths in our argument graph.

In the following, we first present our method of acquiring an argument graph and then describe the acquisition of our argument graph on UK university fees using our method.

2.1 Argument Processing

To address the problems above we opted for using *crowd sourcing* as a means to obtain the arguments for the argument graph. For the first level (i.e. depth 1)

participants are crowd sourced and presented with the root argument in a survey and asked to counter it with a number of arguments. The resulting collection of arguments in depth 1 are all counterarguments to the root argument.

In the following, we describe a pipeline that allows to automatically extract the best arguments from the ones crowd sourced in each depth in order to include them in the graph and collect their counterarguments in the next level.

1. Argument Length We want a potential chatbot to give counterarguments that are neither too short, nor too long. We therefore remove all arguments that are below 15 and above 50 words in length. We would not want a potential chatbot to give a short statement as a counterarguments to the user's argument. We do not include arguments longer than 50 words because these likely contain several arguments and we also do not consider them suitable for a chatbot knowledge base (imagine a chatbot replying with a whole paragraph).

2. Choice of topic words We then extract the most common words from the data (excluding stop words and words that do not add value in the given domain). The definition of *most common* depends on the size and nature of the data and is therefore up to the researcher to decide.

From the most common words, we then select *topic words* which are words which we consider meaningful in the given context. The choice of suitable topic words depends entirely on the domain and their choice is also left to the researchers' discretion. For example, in a set of arguments on university fees, the word *money* appeared many times. It is, however, not very meaningful, whereas the words *debt* and *affordable* tell us more about the topic of the arguments. So by inspecting the frequently occurring words, the researcher can apply their knowledge of the domain to decide which would be good topic words. All arguments that contain at least one topic word are kept, the rest are removed. It should be noted that the list of topic words increases with each depth. The threshold as to how often a word has to appear in order to be considered "common" also rises since the number of arguments increases with each depth.

3. Spell-check We keep all arguments that contain no spelling mistakes. This can be checked by using *Grammarly*[3]. We delete all arguments where Grammarly highlights a typo in order to avoid including arguments into the chatbot knowledge base that contain spelling mistakes since this could influence the persuasive power of the argument. However, we do not consider incorrect punctuation or missing capitalisation as spelling mistakes, given the informality of the setting. Unfortunately, there is no Grammarly API as of the time of writing, and we therefore had to copy-paste the arguments into the Grammarly app.

4. Final Selection of arguments for current depth The arguments that are left after steps 1-3 are presented to crowd sourced participants who are

[3]https://app.grammarly.com/

instructed to select those arguments that they find communicate their message the best. We opted for this wording since we were not interested in the message of the arguments (e.g. its believability or convincingness) but still want to include clear, understandable and appropriate arguments in our graph. The highest-ranked arguments are then included in depth 1 of the argument graph.

Subsequent levels of depth In order to minimise the need for crowd sourcing in Step 4 and in subsequent levels of depth we only keep arguments that covered (i.e. contained) the highest number of topic words. We only present arguments to crowd sourced participants for ranking, where the topic words are the same and a selection has to be made. This ranking is as in step 4 where we ask the participants which arguments communicate their message the best. This way the need for participants in Step 4 is reduced significantly after depth 1. The idea behind this method is to include arguments in the argument graph that address the maximum number of issues as represented by the topic words.

2.2 Argument Acquisition for Next Depth

The arguments for all subsequent levels were collected by presenting the arguments from the previous level to crowd sourced participants who were asked to counter them. Steps 1-3 are then applied to the collected arguments for that level. The participants were presented the last two arguments in the graph since presenting only the last may be confusing without the attacked one as a reference. For example, during the acquisition of arguments in depth 4, participants are shown the argument from depth 2, one of its counterarguments in depth 3 and asked to assume the stance of the argument in depth 2 and counter the argument in depth 3.

3 Case Study and Corpus

In the UK, the current situation is that home students (students from the EU, including the UK) pay around 9000£ tuition fees per year for a Bachelor's degree. This is a controversial situation, with supporters and contestants on both sides. We therefore chose this as a suitable topic for our task and selected *"Universities in the UK should continue charging students the 9k tuition fee per year"* as the root topic for our graph. In the following, the describe how we acquired our argument graph corpus on university fees in the UK applying our method described above.

Participants were recruited via *Prolific*[4], which is an online recruiting platform for scientific research studies, and were paid for taking part in our study. We used Google Forms for our study. The prerequisites for taking part in the study were to be over 18, fluent in the English language and a current resident

[4]https://www.prolific.ac/

of the UK (in order to minimise the risk of recruiting participants who do not know anything about university fee situation in the UK).

For depth 1 we recruited 91 participants who were asked to provide 3 different reasons in a Google Form on why they think that the 9k tuition fees in the UK were not appropriate and should be abolished. We therefore collected 273 (3 x 91) arguments at depth 1.

Many responses consisted of short statements like *"It is too expensive"* or *"students are poor people"* which we would not want a potential chatbot to give as counterarguments to the user's argument. During the argument acquisition in future depths we instructed the participants to provide arguments that were at least 15 words in length as we were only left with 97 arguments after this step in depth 1[5].

We then extracted the most common words from our data. We mentioned above that we delete stopwords and words that do not add value in the given domain from our data. In our case these were words like *education, university, fee, abolish, students, degree* and *tuition*. We extracted all words that came up at least 5 times in the dataset of 97 arguments.

From the most common words we selected the words *job, debt, afford/ affordable, access/accessible* and *free* as topic words for depth 1. Other common words included *study, high, amount, money, pay* and *work*, which we believed were too general. We mentioned above that the list of topic words grows with each depth: In depth 2, for example, the words *loan, tax, government* and *scholarship* were added to the list of topic words.

After steps 1-3 we were left with 48 arguments out of the 273 at depth 1. At depth 1 we decided to include 3 arguments for each topic word in the graph. We created 5 surveys (one for each topic word) which presented all arguments that included the topic word in question. We crowd sourced 20 participants per survey and instructed them that the arguments might be very similar and all touch on a certain aspect but that the individual arguments differ in their quality. We asked them to select those arguments that they found communicate their message the best. We then used the three arguments that were ranked the highest in each group (some arguments contained two topic words, therefore some topic words are represented by more than 3 arguments).

Our aim was to create a graph where each argument after depth 1 has 3 counterarguments (on average) to avoid making the graph too big and due to limited funding. In subsequent depths we only kept arguments that covered the highest number of topic words. Only if the topic words of several arguments were the same and a selection had to be made those arguments were presented to crowd sourced participants for ranking.

For example, consider an argument in depth 1 that had 6 counterarguments in depth 2 after applying Steps 1-3. The counterarguments (CA) contained the following topic words:

[5]When the study took place Google Forms did not support response validation. Since July 2019 a minimum character count can be specified.

CA 1 loan, debt
CA 2 loan, debt, scholarship
CA 3 loan, government
CA 4 government
CA 5 loan
CA 6 loan, government

CA 1 and CA 2 were selected for the next depth because they contained the highest number of topic words and no other CAs contained the same combination. CA 3 and CA 6 were presented in a survey to participants in order to choose the "better" one for the graph. CA 4 and CA 5 were not selected because all other CAs contained at least one of the topic words of CA 4 and CA 5.

Depth 1 consists of 16 arguments. We created 3 surveys (containing of 5, 5 and 6 arguments respectively) and recruited 10 participants per surveys to counter the given arguments. We split the arguments into three smaller surveys in order to avoid presenting similar arguments and reduce the risk of participants giving the same counterargument to several arguments. For each subsequent level of depth, the arguments from the previous depth were divided into surveys of 5-6 arguments and 10 participants were recruited per survey. We therefore acquired 10 counterarguments per argument in each depth. Participants were presented the last two arguments in the graph. For example, during the acquisition of arguments in depth 4, participants were shown the argument from depth 2 (against fees), one of its counterarguments in depth 3 (pro fees) and asked to assume the position of being against fees and counter the argument in depth 3. It should be noted that for depth 5 we only recruited 5 participants to counter the arguments of depth 4.

3.1 The Corpus

Our graph contains 1288 arguments with each argument on average having 3 counterarguments, and consists of 5 depths making it the most extensive corpus of this kind. The overall corpus of acquired arguments contains over 4000 arguments.

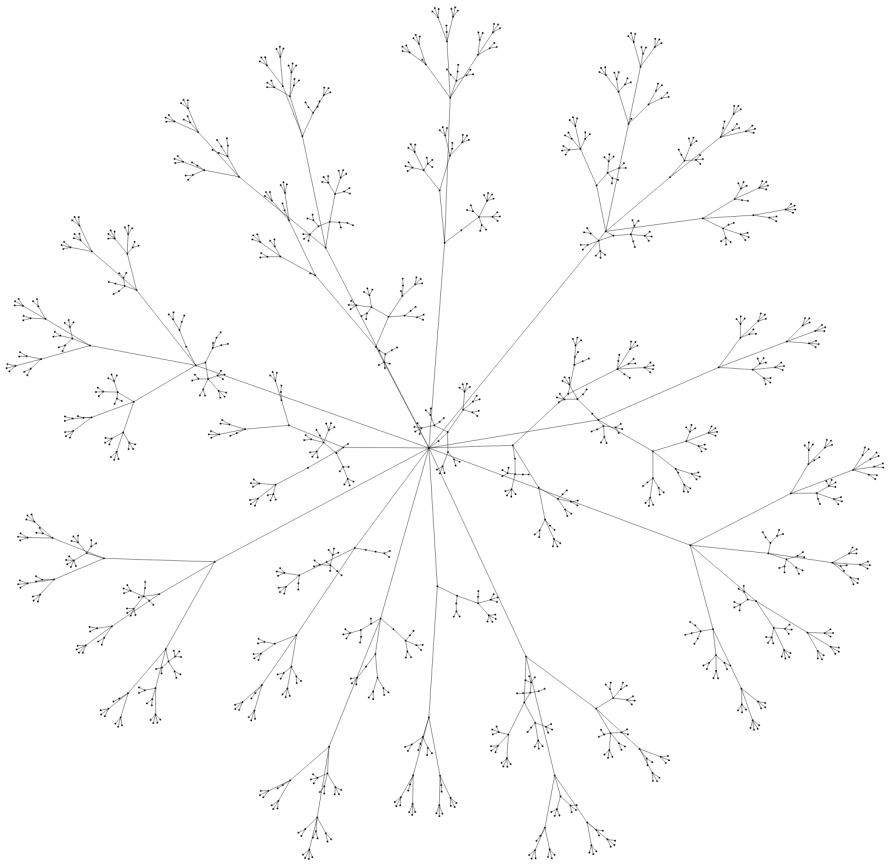
The generated corpus can be found on github [21]. It consists of two data sets. One data set contains the raw arguments acquired for each depth. The second dataset contains the arguments that were used in the generation of the argument graph. Each argument contains a unique ID and the ID of the attacked argument in the previous depth. For example, an argument in depth 2 may have the id *depth2_6* and the ID of the attacked argument *depth1_34* which means argument *depth2_6* attacks argument *depth1_34*.

The github repository also contains the `python` code to generate a visual network graph using the `pyvis` library. The resulting visualisation displays the arguments when hovering over the nodes and is shown in Figure 3 (a higher resolution picture is available in the github repository) [21].

4 Evaluation

We evaluated our generated argument corpus by randomly creating 24 dialogues by following the arcs of the graph, starting from the root and following each of

Fig. 3. Visualisation of the generated argument corpus in graph form



the 16 arcs from the root to the argument in depth 1 at least once. This way we ensured to create at least 16 completely distinct dialogues. We divided the 24 dialogues into 4 surveys using Google Forms and recruited 20 participants for each survey to judge the 6 given dialogues. An example dialogue is given below.

PERSON A: *Universities in the UK should continue charging students the 9k tuition fee per year.*

PERSON B: *Education should be available for everyone, not for only ones who can afford it.*

PERSON A: *People who can't afford have government help. Government can't afford free education for all unless they increase the taxes and people won't like it.*

PERSON B: *The government are still paying for the loans and probably won't see the money back when the loans are written off in 30 years time. Cheaper education and higher taxes is more sustainable than relying on students to pay back the loans, which they won't.*

PERSON A: *The government should step out then and leave it to the banks to take the risk. Anyway with higher taxes and cheap education there would be plenty of educated unemployed to pay by the Government.*

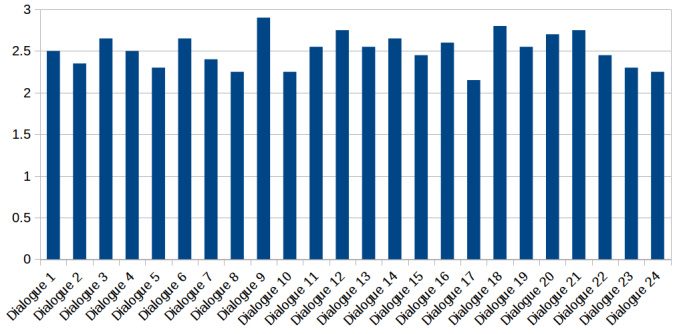
PERSON B: *Banks would likely impose even higher rates of interest which would be unsustainable and they may also reject a large number of students given their financial circumstances.*

We informed the participants that the study involved judging transcripts and that the given dialogues involved two parties arguing whether tuition fees in the UK should be kept at 9000£. Party A believes they should be kept and Party B believes they should be abolished. We instructed them to judge 6 transcripts plus an additional one playing the role of an attention check to ensure honesty/quality of the provided responses. We asked them to score the transcripts in respect of each party staying to the point and defending their point of view. We asked them to not judge the dialogues by whether they believed the presented arguments since we were only interested in the overall quality of the dialogue (whether they make sense and parties sticking to their point of view). The participants were given a choice of three:

1. *Both parties don't stick to the point and don't defend their point of view*
2. *Both parties somewhat stick to the point and somewhat defend their point of view*
3. *Both parties do stick to the point and do defend their point of view*

On average each dialogue scored 61% for option 3 (both parties sticking to the point and defending their point of view), 29% for option 2, and only 10% for option 1. Figure 4 shows the score for each dialogue, option 1 (*don't*) receiving score 1, option 2 (*somewhat*) receiving score 2 and option 3 (*do*) receiving score

Fig. 4. Scores for each individual dialogue.



3. The average score per dialogue was 2.51 which shows that the dialogues were of good quality and that if following a path in the graph, the resulting dialogue makes sense despite the individual arguments being collected from different people.

5 Discussion and Conclusions

In this paper, we introduce a methodology to acquire a corpus of arguments for dialogues and present a corpus for research for computational argumentation, natural language processing, and chatbot knowledge base construction. Apart from checking for spelling mistakes, we have not conducted any further quality assessment of the arguments and have not checked for duplicate arguments in the argument graph. This gives researchers the possibility to use our corpus for research in methods like:

- Argument similarity assessment [22, 23]: many arguments in the graph support the same idea and are fairly similar. However, one can say the same thing in completely different ways, and clustering arguments by their similarity is a challenging but potentially valuable task.
- Argument quality assessment [24–26]: After clustering similar arguments together one could apply some sort of quality assessment in order to decide which argument in the cluster is the "best" according to some criteria (e.g. convincingness [27]).
- Establishing more attack (and support) relationships between arguments in the graph [28, 29]: After identifying similar arguments one could establish more attack relationships in the graph. For example, if arguments A and B are the same (just differently phrased), the counterarguments of A also attack B and vice versa.

By applying the methods above high-grade chatbot knowledge bases could be created that contain only arguments of the highest quality (however one chooses to assess that) and contain a high number of possible arguments for that domain.

We also evaluated the quality of our corpus and believe that publishing it will give researchers a resource to explore the topics mentioned above, which will facilitate further research in these areas.

In future work we want to create a chatbot that uses our generated argument graph as knowledge base and use it in a study with real participants. The participants could be on either side of the debate (either for of against keeping university fees) and the chatbot would defend the opposite standpoint. In order to evaluate our chatbot, the participants could be asked to judge the chat with the chatbot on persuasiveness and other metrics like the quality of the dialogue and whether the chatbot gave relevant replies (counterarguments).

6 Acknowledgments

The first author is funded by a PhD studentship from the EPSRC. The authors would like to thank Sylwia Polberg for valuable feedback on earlier versions of this paper.

References

1. A. Hunter. Computational persuasion with applications in behaviour change. In *Proc. of Computational Models of Argument 2016*, pages 5–18, 2016.
2. P.M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.
3. P. Besnard, A. Javier Garca, A. Hunter, S. Modgil, H. Prakken, G. Simari, and F. Toni. Introduction to structured argumentation. *Argument and Computation*, 5(1):1–4, 2014.
4. P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. In *Knowledge Engineering Review 26(4)*, pages 365–410, 2011.
5. A. Hunter, L. Chalaguine, T. Czernuszenko, E. Hadoux, and S. Polberg. Towards computational persuasion via natural language argumentation dialogues. In *Proc. of Kuenstliche Intelligenz 2019 (in press)*, 2019.
6. Reddit thread. https://tinyurl.com/y267p2lq.
7. J. Huang, M. Zhou, and D. Yang. Extracting chatbot knowledge from online discussion forums. In *Proc. of the 20th International Joint Conference on Artificial Intelligence*, pages 423–428, 2007.
8. D. Toniuc and A. Groza. Climebot: An argumentative agent for climate change. In *Proc. of the 2017 IEEE 13th International Conference on Intelligent Computer Communication and Processing*, pages 63–70, 2017.
9. L. A. Chalaguine, A. Hunter, F. L. Hamilton, and H. W. W. Potts. Impact of argument type and concerns in argumentation with a chatbot. In *Proc. of the 31st International Conference on Tools with Artificial Intelligence 2019 (in press)*, 2019.
10. S. Wells. Argument mining: Was ist das? In *Proc. of the 14th International Workshop on Computational Models of Natural Argument*, 2014.
11. E. Cabrio and S. Villata. Five years of argument mining: a data-driven analysis. In *Proc. of the 27th International Joint Conference on Artificial Intelligence*, pages 5427–5433, 2018.
12. M. Lippi and P. Torroni. Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology 16(2)*, pages 1–25, 2016.
13. I. Habernal and I. Gurevych. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179, 2017.
14. R. Swanson, B. Ecker, and M. Walker. Argument mining: Extracting arguments from online dialogue. In *Proc. of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 217–226, 2015.
15. A. Pazienza, S. Ferilli, and F. Esposito. Constructing and evaluating bipolar weighted argumentation frameworks for online debating systems. In *Proc. of the 1st Workshop on Advances In Argumentation In Artificial Intelligence*, pages 111–125, 2017.

16. C. Tan, V. Niculae, C. Danescu-Niculescu-Mizil, and L. Lee. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proc. of the 25th International Conference on World Wide Web*, pages 613–624, 2016.
17. R. Levy, B. Bogin, S. Gretz, R. Aharonov, and N. Slonim. Towards an argumentative content search engine using weak supervision. In *Proc. of the 27th International Conference on Computational Linguistics*, pages 2066–2081, 2018.
18. E. Hadoux and A. Hunter. Comfort or safety? Gathering and using the concerns of a participant for better persuasion. 2019.
19. A. Hunter, S. Polberg, and E. Hadoux. Strategic argumentation dialogues for persuasion: Framework and experiments based on modelling the beliefs and concerns of the persuadee. Technical report.
20. T. Krauthoff, C. Meter, and M. Mauve. Dialog-based online argumentation: Findings from a field experiment. In *Proc. of the 1st Workshop on Advances in Argumentation in Artificial Intelligence*, pages 85–99, 2017.
21. Corpus: https://github.com/lisanka93/Argument_Graph_Corpus.
22. F. Boltuzic and J. Snajder. Identifying prominent arguments in online debates using semantic textual similarity. In *Proc. of the 2nd Workshop on Argumentation Mining*, pages 110–115, 2015.
23. A. Misra, B. Ecker, and M. A. Walker. Measuring the similarity of sentential arguments in dialog. In *Proc. of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 276–287, 2016.
24. H. Wachsmuth, S. Syed, and B. Stein. Retrieval of the best counterargument without prior topic knowledge. In *Proc. of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 241–251, 2018.
25. H. Wachsmuth, N. Naderi, I. Habernal, Y. Hou, G. Hirst, I. Gurevych, and B. Stein. Argumentation quality assessment: Theory vs. Practice. In *Proc. of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 250–255, 2017.
26. H. Wachsmuth, N. Naderi, Y. Ho, Y. Bilu, V. Prabhakaran, A. T. Thijm, G. Hirst, and B. Stein. Computational argumentation quality assessment in natural language. In *Proc. of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 176–187, 2017.
27. I. Habernal and I. Gurevych. Which argument is more convincing? Analyzing and predicting convincingness of web arguments using bidirectional LSTM. In *Proc. of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1589–1599, 2016.
28. O. Cocarascu and F. Toni. Identifying attack and support argumentative relations using deep learning. In *Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1374–1379, 2017.
29. L. A. Chalaguine, A. Hunter, F. L. Hamilton, and H. W. W. Potts. Argument harvesting using chatbots. In *Proc. of Computational Models of Argument 2018*, pages 149–160, 2018.