Audience-Aware Argument Generation

From the
Faculty of Computer Science, Electrical Engineering, and Mathematics
of the
University of Paderborn

The submitted dissertation of
Milad Alshomary
to obtain the academic degree of
Dr. rer. nat.

Paderborn, Germany
September 2023

Dissertation

Audience-Aware Argument Generation
Milad Alshomary, Paderborn University
Paderborn, Germany, 2023

Reviewers

Prof. Dr. Henning Wachsmuth, Leibniz University Hannover
Prof. Dr. Ivan Habernal, Paderborn University
Asst. Prof. Dr. Joonsuk Park, Richmond University

Doctoral Committee

Prof. Dr. Henning Wachsmuth, Leibniz University Hannover
Prof. Dr. Ivan Habernal, Paderborn University
Asst. Prof. Dr. Joonsuk Park, Richmond University
Jun. -Prof. Dr. Martin Potthast, Leipzig University
Prof. Dr. Carsten Schulte, Paderborn University

To my family and loved ones

Contents

Preface

Abstract

Driven by the advances of machine learning, the field of computational argumentation witnessed significant progress that has led to various applications, like argument search engines, argumentative writing assistants, and debate technologies. The latter aims to bring new and diverse aspects to discussions by providing arguments taking the opposite stance of a human opponent and countering their arguments. This task can encourage critical thinking, widen the audience's perspective on controversial topics, and maximize agreement among them. However, to achieve this goal, debate technologies have to overcome several challenges, such as synthesizing natural language arguments that are (1) relevant to the discussed topic, (2) addressing their user's beliefs, and (3) accounting for opposing arguments. Our work aims to contribute methods and insights to advance research towards overcoming these challenges. Accordingly, our main research question is *how to achieve more effective engagement of debate technologies?* To answer this question, we study each of the three mentioned challenges and propose approaches to address them.

In particular, we first devise a conceptual architecture of a debate technology framework and explain how one can use our contributions to form a single approach to debate technology. We then focus on each of the three main tasks in detail. First, to identify the discussion space, we propose an approach that takes a collection of arguments and distills a set of key points by first ranking sentences in terms of their key point candidacy, then aggregating them into a final set that represents the discussion space. To model key point candidacy, our approach infers the importance of a sentence from its centrality in the collection and argumentativeness. Our experiments show that our approach achieves strong results in generating representative key points. Second, to model the audience, we propose two alternative approaches. One approach uses the audience's stances on big issues to build a model of their belief, while the other uses their moral foundations. We then propose methods to use these belief models to guide argument generation to generate natural language arguments that address the corresponding audience model of beliefs. We empirically show that our approach can encode the audience's beliefs into natural language arguments and demonstrate that morally framed arguments

are more effective than generic ones. Finally, to counter opposing arguments, we study the task of inferring the main point (conclusion) and the weak premises of an argument. We then use this inferred knowledge to generate more effective and relevant counters. Our empirical results show a boost in the effectiveness of generating counters when considering this inferred knowledge.

Our research findings indicate the relevancy of the studied tasks to the overall debate task. The identified discussion space can assess the debate technology in synthesizing relevant arguments. By constructing a model of its audience's beliefs, debate technology can generate more empathetic arguments that increase agreement among its audience. Finally, by identifying weak points in its opponent's argument, the debate technology can synthesize more relevant counters that increase its audience's trust.

Zusammenfassung

Angetrieben von den Fortschritten des maschinellen Lernens hat das Gebiet der rechnergestützten Argumentation erhebliche Fortschritte gemacht, die zu verschiedenen Anwendungen wie Argument-Suchmaschinen, Schreibassistenten und Debattentechnologien geführt haben. Letztere haben zum Ziel, neue und vielfältige Aspekte in Diskussionen einzubringen, indem sie Argumente vorbringen, die die entgegengesetzte Position eines menschlichen Kontrahenten vertreten und dessen Argumente entkräften. Diese Anwendung kann das kritische Denken fördern, die Perspektive des Publikums zu kontroversen Themen erweitern und die Zustimmung unter ihnen maximieren. Um dieses Ziel jedoch zu erreichen, müssen Debattentechnologien mehrere Herausforderungen bewältigen, wie zum Beispiel die Synthese von natürlichsprachigen Argumenten, die (1) relevant für das diskutierte Thema sind, (2) die Überzeugungen ihrer Benutzer ansprechen und (3) gegnerische Argumente berücksichtigen. Unsere Arbeit zielt darauf ab, Methoden und Erkentnisse zur Forschung an diesen Herausforderungen beizutragen. Entsprechend lautet unsere Hauptforschungsfrage: "Wie kann eine effektivere Einbindung von Debattentechnologien in Debatten erreicht werden?" Um diese Frage zu beantworten, untersuchen wir jede der drei genannten Herausforderungen und schlagen Ansätze zu ihrer Bewältigung vor.

Wir entwickeln zunächst eine konzeptionelle Architektur eines Debattentechnologie Frameworks und erläutern, wie unsere Beiträge verwendet werden können, um einen Ansatz zur Debattentechnologie zu entwickeln. Anschließend konzentrieren wir uns detailliert auf jede der drei Hauptaufgaben. Erstens, um den Diskussionsraum zu identifizieren, schlagen wir einen Ansatz vor, der eine Sammlung von Argumenten betrachtet und eine Reihe von Schlüsselpunkten heraushristaslisiert, indem Sätze zunächst hinsichtlicher ihrer Eignung als Schlüsselpunkt bewertet werden und sie dann zu einem endgültigen Satz aggregiert, der den Diskussionsraum repräsentiert. Um die Eignung als Schlüsselpunkt zu modellieren, schließt unser Ansatz die Bedeutung eines Satzes aus seiner Zentralität in der Sammlung von Argumenten und wie argumentativ es ist. Unsere Experimente zeigen, dass unser Ansatz starke Ergebnisse bei der Erzeugung repräsentativer Schlüsselpunkte erzielt. Zweitens schlagen wir zwei alternative Ansätze vor, um das Publikum zu modellieren. Ein Ansatz verwendet die vom Publikum vertetener Positionen zu großen Themen, um ein Modell ihrer Überzeugungen zu erstellen, während der andere ihre moralischen Grundlagen verwendet. Wir schlagen dann Methoden vor, um die Generierung von natürlichsprachlichen Argumenten zu leiten, die eine spezifische Überzeugung des Publikums ansprechen. Wir zeigen empirisch, dass unser Ansatz die Überzeugungen des Publikums in natürlichsprachliche Argumente kodieren kann, und stellen fest, dass moralisch formulierte Argumente effektiver sind als generische. Schließlich, um Argumente der Gegenseite zu entkräften, untersuchen wir die Aufgabe, die Hauptaussage (Schlussfol-

gerung) und die schwachen Prämissen eines Arguments abzuleiten. Wir verwenden dieses abgeleitete Wissen, um effektivere und relevantere Gegenargumente zu generieren. Unsere empirischen Ergebnisse zeigen eine Steigerung der Effektivität bei der Generierung von Gegenargumenten, wenn dieses abgeleitete Wissen berücksichtigt wird.

Unsere Forschungsergebnisse zeigen die Relevanz der untersuchten Aufgaben für die Gesamtaufgabe der Debattierung. Der identifizierte Diskussionsraum kann die Debattentechnologie bei der Synthese relevanter Argumente bewerten. Durch die Konstruktion eines Modells der Überzeugung des Publikums kann die Debattentechnologie einfühlsamere Argumente generieren, die die Zustimmung im Publikum erhöhen. Schließlich kann die Debattentechnologie durch die Identifizierung von Schwachstellen im Argument des Kontrahenten relevantere Gegenargumente synthetisieren, was das Vertrauen des Publikums steigert.

Acknowledgment

I want to express my profound gratitude to my supervisor, Prof. Henning Wachsmuth, for his support, guidance, and mentorship throughout my doctoral journey. Your expertise, insightful feedback, and hours of discussions have been instrumental in shaping my research views in general and this thesis in particular. I am truly fortunate to have had the opportunity to work under your supervision.

I would also like to thank Prof. Ivan Habernal and Asst. Prof. Joonsuk Park for taking the time to review my work and giving me valuable feedback. I thank Jun.-Prof. Martin Potthast and Prof. Carsten Schulte for being part of my doctoral committee.

I sincerely appreciate my beloved wife, Sarah, for her support and understanding. Your patience, encouragement, and love sustained me through the challenging moments of this academic journey.

This work was conducted first in the Computational Social Science (CSS) department at Paderborn University and later at the Natural Language Processing (NLP) department at the Institute of Artificial Intelligence at Lebniz University Hannover. Throughout this time, I collaborated with brilliant colleagues, mentioned in chronological order of my interaction with them: Michael Völske, Yamen Ajjour, Zahra Nouri, Shahbaz Syed, Wei-Fan Chen, Timon Ziegenbein, Max Spliethöver, Roxanne El Baff, Johannes Kiesel, Maja Stahl, and Meghdut Sengupta, and Gabriella Skitalinska. Thanks for the intellectual discussions, fun times, and shared experiences that enriched my doctoral journey.

Chapter 1

Introduction

Argumentation is at the core of human communication. It is the means by which humans resolve conflict and reach an agreement. Throughout history, argumentation has drawn much attention and become the focal point for many scholars. Recent research in Natural Language Processing (NLP) has led to several advances in analyzing and synthesizing natural language arguments, which resulted in several practical end-user systems. Among them are debate technologies that can engage in human debates. An essential feature of these debate technologies is the ability to synthesize new arguments on a specific topic or counter other proposed arguments in the debate. Nevertheless, research on the effective generation of arguments in natural language is still under-explored, partly due to the adversity of this task. Synthesizing effective arguments requires a set of abilities such as commonsense reasoning, rhetoric, and empathetic understanding, which remain to this day challenging aspects for the machine to master.

Our work aims to advance research on argument generation by identifying aspects of effective arguments in debates and proposing methods to address each. In particular, an effective argument should be relevant to the discussion, consider the opponent's argument, and address the audience's interests. Hence, throughout this thesis, we propose methods to extract knowledge about relevant key points to the discussion, the main point of the opponent's argument and its weak premises, and a model of the audience's belief. We then utilize this knowledge to guide the generation of more effective and empathetic (counter) arguments. This chapter will start with a background on argumentation and the recent advances in computational argumentation (Section 1.1 and 1.2). We will then highlight the limitations of current related work concerning the aspects above (Section 1.3). Finally, we will introduce our proposed methods to circumvent these shortcomings and discuss the key finding of this research (Section 1.4).

1.1 Argumentation

Argumentation is omnipresent in our lives. People have engaged in argumentation dialogues throughout history, from free-style argumentation on dinner tables to formal debates on big podiums. In ancient Greece, debates took the name of *sophistical refutations* (Hasper, 2013), where two sides, the questioner and the answerer, engage in a discourse to uncover the truth of a put-forward thesis. In medieval Europe, *Scholastic disputation* (Novikoff, 2012) rose as a method to investigate questions in science and theology. The twentieth century witnessed the growth of informal logic as a research field attempting to analyze argumentation as it occurs in daily life to enhance one's critical thinking.

Through argumentation, we can reach an agreement with others and acquire new perspectives on controversial issues to make more informed decisions. However, resolving a disagreement between disputed parties through argumentation is a hard endeavor, even for humans. Hence, several thinkers throughout history dedicated their research to understanding and analyzing argumentation. They provided tools and theories on what makes argumentation successful in their goal. From early history, Aristotle argued that arguments draw their strength from being reasonable (logos), appealing to emotions (pathos), or emphasizing the credibility of their author (ethos) (Aristotle and Kennedy, 2006). In the New Rhetoric, Perelman (1971) emphasized that since argumentation aims to influence a specific group, its content and approach should be subject to the beliefs and characteristics of such a group. To van Eemeren and Houtlosser (1999), when engaging in a debate, one might construct their argument based on three aspects: (1) relevant discussion points to be addressed, (2) characteristics of the targeted audience, and (3) appropriate style of presentation. Walton (2009) argues that a counter to an argument can be an attack on its conclusion (rebuttal), the validity of reasoning of its premises toward its claim (undercut), or the validity of one of its premises (undermining).

Motivated by the importance of argumentation, our work draws inspiration from the theories above to propose effective methods for argument generation. For example, we follow van Eemeren and Houtlosser (1999) in considering the importance of identifying the topic's discussion space and the audience's characteristics when engaging in a debate. When countering a given argument, we aim to model the argument undermining and rebuttal phenomena mentioned by Walton (2009).

1.2 Computational Argumentation

The popularizing of online forums and social media opened up spaces for everyone to engage in daily argumentation, resulting in risks of misinformation, polarization, and echo chambers (Cinelli et al., 2021). This situation demands computational tools to assess humans in widening their perspective on controversial topics by providing a diverse set of arguments on the subject, questioning their views, and

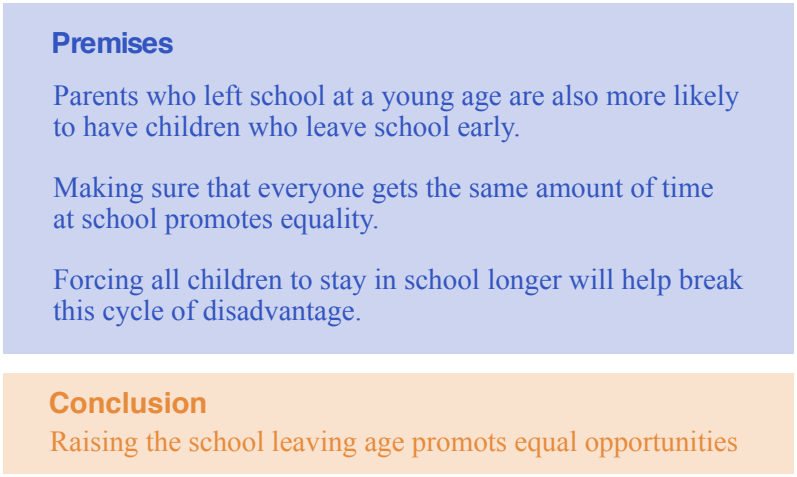

FIGURE 1.1: An example argument taken from the student persuasive essays dataset (Stab and Gurevych, 2014b) including a set of premises and the corresponding conclusion.

collecting evidence for a particular claim to fight misinformation. The Computational Argumentation field (CA) is one of the leading research areas that investigate approaches to build these tools, and it is the field to which this thesis belongs.

The CA field investigates methods to enable machines to mine and synthesize arguments in natural language texts (Stede and Schneider, 2018). While there exist some theoretical models of argumentation (Toulmin, 1958), in practice, CA approaches often simplify a natural language argument into a composition of premises that reason towards (*pro stance*) or against (*con stance*) a main point called the argument's conclusion. Figure 1.1 presents an example argument consisting of a set of premises with a pro stance towards the conclusion *Raising the school leaving age promotes equal opportunities*. These premises can be categorized into different types, such as statistics, anecdotes, and assumptions (Al-Khatib et al., 2016). Typically, CA approaches to mine arguments takes a text as an input and output a representation of either the internal structure of a single argument (premises, conclusion) (Stab and Gurevych, 2014b) or the external interaction between multiple arguments in a discussion (Chakrabarty et al., 2019b). Argument synthesis includes the generation of either a missing argument component, an argumentative text on a topic, or a counter-argument. Besides these two main branches of CA, argument quality has also been heavily studied in the community (Wachsmuth et al., 2017a), with approaches either addressing specific quality dimensions (Stab and Gurevych, 2017) or following a holistic approach in assessing the quality of an argument (Gretz et al., 2020b).

The field of CA has benefited from the advances in natural language processing (NLP) and machine learning (ML). Early approaches focused on hand-crafted features to model various argument-mining tasks, while the absence of big corpora

and the limitation of computational power restricted the exploration of argument-generation tasks. But, recently, the rise of transfer learning and the invention of powerful deep neural networks allowed a new era of text generation (Radford et al., 2019). The community proposed several end-to-end neural architectures that learn to generate argumentative texts (Hidey and McKeown, 2019, Hua and Wang, 2018) and demonstrated a promising future. However, with great power comes more responsibilities. The new era also opened up a host of challenges and societal concerns, such as the explainability of neural models (Danilevsky et al., 2020), the faithfulness of generated texts (Li et al., 2022), and bias in all its forms (Sheng et al., 2021). Addressing each of these challenges is an active research field on its own. For example, ensuring faithfulness of generated texts can be tackled through utilizing external factual knowledge about the world and ensuring that generated text adheres to this knowledge (Dinan et al., 2018, Shuster et al., 2021). Due to its subjective nature, argumentation language might contain different forms of bias, which appear in trained models (Spliethöver and Wachsmuth, 2020). A line of CA research attempts to build argumentation models that are fair and free from bias (Holtermann et al., 2022). Throughout this thesis, we will highlight, whenever possible, our research's societal impact and concerns and how to potentially deal with them.

Automating the process of argument analysis and generation enable a host of beneficial end-user applications and tools like argument search engines (Wachsmuth et al., 2017b, Daxenberger et al., 2020), decision-making assistants (Costa et al., 2017), and debate technologies (Slonim et al., 2021). *Debate technologies* are systems that can either assess humans in a discourse dialogue or fully take the role of a human debater autonomously. Such task is not straightforward since it requires a set of abilities, such as the synthesis of natural language arguments. These assistants could play an important role in solving disagreements between engaged users in a debate by learning and focusing on the common concerns of its users rather than on disputes. Recently, Slonim et al. (2021) proposed a system, called *Project Debater*, that decomposes the overall debating task into a set of subtasks. These subtasks include argument mining and indexing, principle argument construction and matching, argument rebuttal, and debate construction, where the final argumentative text is synthesized. The system's performance was demonstrated in a live debate against a human expert in front of an audience[1] and showed a promising future. Our work studies the argument generation task in a debate context, that is, how to improve the effectiveness of generated arguments when synthesized by a debating technology. The rest of this chapter will discuss in detail the importance and challenges of this task and highlight our approach to address these challenges.

[1] https://www.ibm.com/blogs/research/2019/02/ai-debate-recap-think-2019/

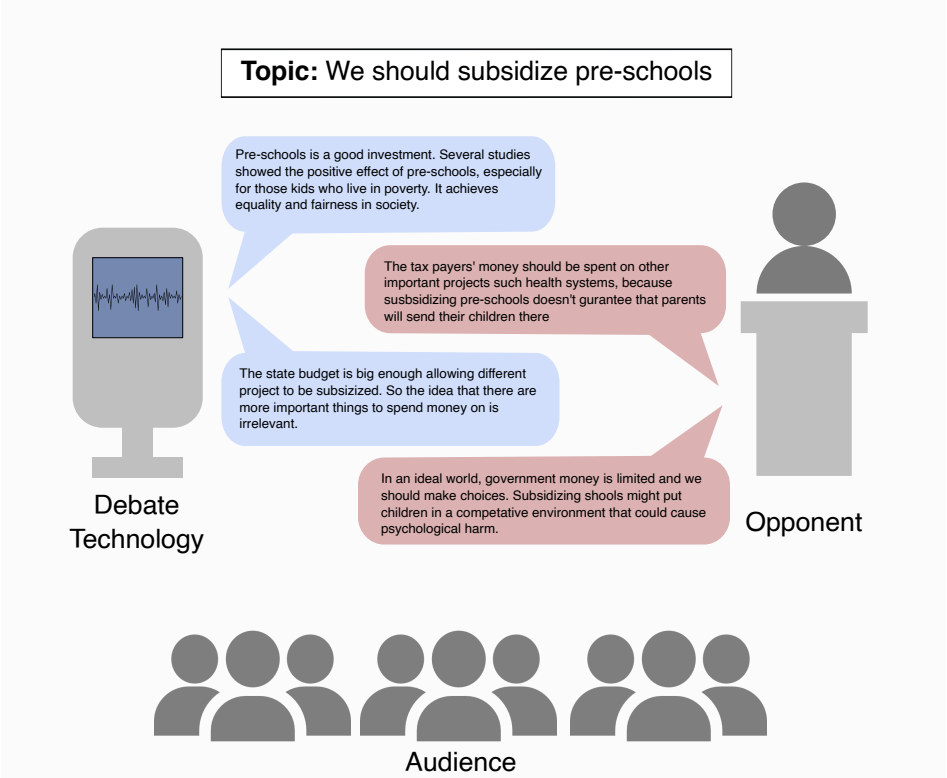

FIGURE 1.2: Debate scenario with two parties (humans or machines) participating in a discussion on the topic *"We should subsidize pre-schools"*. Taking turns, each side puts forward pro/con arguments on the topic. By the end of the discussion, the goal is to increase the agreement between the audience and the debate parties.

1.3 Shortcomings of Argument Synthesis

Despite the potential success demonstrated in the computational synthesis of natural language arguments, there are still under-explored areas to be studied to further increase their effectiveness. To highlight these areas, we will address the argument synthesis task in a debate context with the goal of maximizing agreement among the audience. In particular, as illustrated in Figure 1.2, given a controversial topic phrased as a claim, two agents (a human and an AI) engage in a debate in front of an audience. In this debate, each side puts forward an argument and can provide another argument in which they refute and address points in their opponent's argument. To effectively engage in such a debate, we argue that the technology should handle the following tasks:

- *Modeling the discussion space*, that is, the relevant key points to the given controversial topic that frame the discussion.

- *Modeling the audience's beliefs*, that is, to understand the audience's beliefs and demands in order to achieve a better reach.

- *Modeling the opponent's argument* in terms of the main point (conclusion) put forward by the argument and how to counter it.

In the following, we will discuss in detail each aspect, including challenges and shortcomings, to highlight our contributions and key findings in the following section.

1.3.1 Modeling Discussion Space

The first task the debate participants will have to solve is collecting relevant talking points to use in their arguments. For debate technology, analyzing the discussion space involves retrieving relevant argumentative texts from the web and using them to distill salient points. While it is potentially practical to collect an enormous amount of arguments pertinent to the topic, the extraction of concise statements representing the main points of a discussion is not well studied compared to other genres like news articles, possibly due to the nature of argumentative texts that is implicit and relying much on commonsense and assumptions. Early research on discussion summarization simplified the task to only extracting the mentioned aspects of a discussion and the stance towards them (Egan et al., 2016), ignoring the reasoning component. Moreover, applying general summarization approaches to the task (Erkan and Radev, 2004) might result in extracting statements that are potentially central but not argumentative or do not reflect the core opinion of the argument. Therefore, in this work, we assume a given collection of relevant argumentative texts and focus on summarizing the key points appearing in them. A debate technology can then use these extracted key points as relevant aspects reflecting the topic potential introduced by van Eemeren and Houtlosser (1999) to guide the argument generation process.

Until recently, Friedman et al. (2021a) proposed to study key point analysis as a shared task at the 8th Workshop on Argument Mining located at the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021). Given a collection of arguments in natural language, the task is to extract and aggregate concise argumentative statements that summarize the collection and match them to their corresponding arguments. The authors propose preliminary approaches to this task that use basic rule-based techniques to generate candidate key points based on their argumentative quality and then select the ones with high scores in matching input arguments (Bar-Haim et al., 2020b). In our research, we explore more sophisticated approaches to model key points. These approaches consider, besides argumentativeness, the centrality of sentences in the argument collection.

1.3.2 Modeling Audience's Beliefs

The audience plays an essential role in argumentation. As mentioned, according to van Eemeren and Houtlosser (1999), the audience's demand is an important factor that dictates the next argumentation move to be performed in a debate. For example, the debater might focus on shared human values with their audience and avoiding unnecessary conflicts. So far, research on computational argumentation has focused on studying argument persuasiveness and its correlation with the audience (El Baff et al., 2018, Durmus and Cardie, 2018). Argumentation frameworks like the work of Bench-Capon et al. (2002) integrate the human value preferences of a specific audience as a factor in deciding the acceptance of an argument. Although these works provide evidence of the importance of the audience for persuasion, no methods were proposed that synthesize natural language arguments accounting for the audience's beliefs. Moreover, In social psychology, a line of research studies how people adhere to different moral systems when making judgments on controversial issues (Haidt, 2012). A subsequent line of research demonstrated that one should consider the audience's moral system (Feinberg and Willer, 2015) to craft compelling arguments targeting this audience. In our debate scenario (Section 1.3), we hypothesize that debate technology can generate more effective arguments by using such knowledge about their audience. To this day, computational argumentation research has not studied the tuning of generated arguments toward a targeted audience or discussed models that represent an audience's belief. In the following section, we will present our contribution toward modeling and encoding the audience's beliefs into the process of argument generation.

1.3.3 Modeling Opponent's Argument

Relevant and effective synthesis of counter-arguments is crucial in debates. As mentioned, Walton (2009) states that a counter-argument is an attack on a specific argument that can be either a rebuttal, undermining, or undercutting. Accordingly, analyzing the argumentative structure of a natural language argument is important. One must identify the argument's conclusion and its weak premise(s) to decide what type of counter-argument to produce and its content. Research in computational argumentation addressed the counter-argument synthesis task through either combining and arranging a set of argument units retrieved from an index (*retrieval-based approaches*) or by generating from scratch (*generation-based approaches*).

On the one hand, retrieval-based approaches like the work of Wachsmuth et al. (2018b) and Orbach et al. (2019) rely on a predefined collection of argumentative units (full arguments), which are collected in an offline stage. Once provided with an argument, a matching process retrieves the best relevant counter-argument. Moreover, Bilu et al. (2019) proposed an approach to creating a knowledge base of principled common-place arguments, which rely on first principles and can be

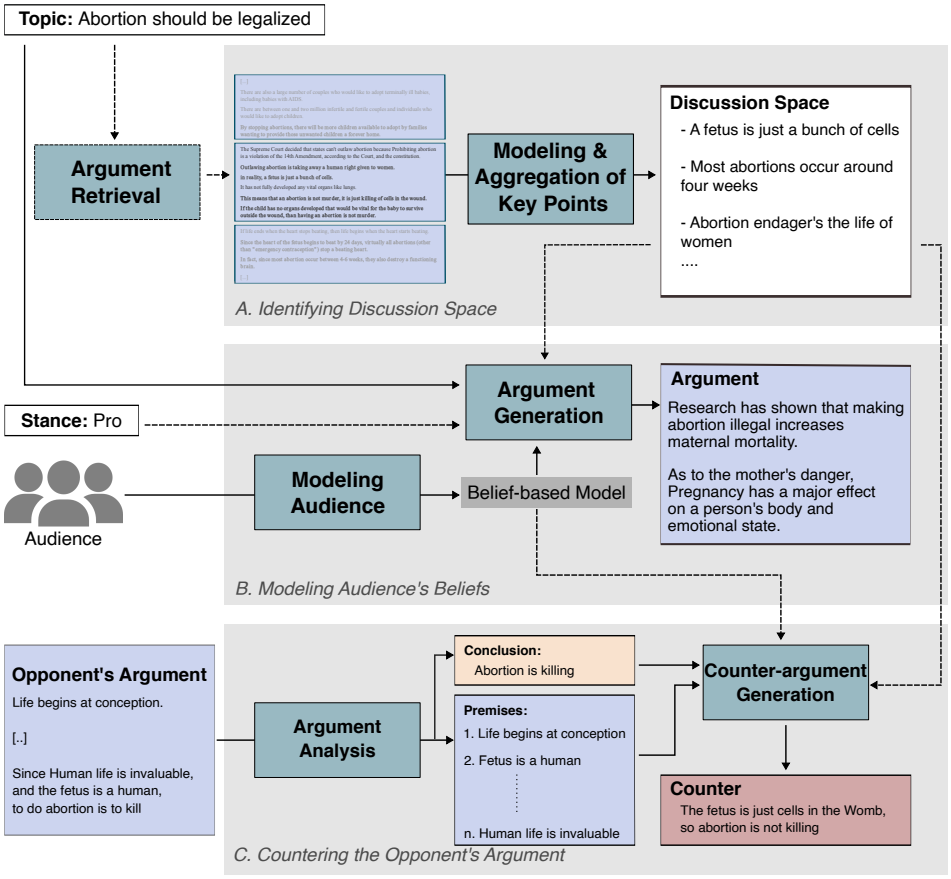

FIGURE 1.3: An overview of our conceptual approach towards effective engagement in Debates that takes a controversial topic, a stance, an audience representation, and the opponent's argument as input. First (A), our approach analyzes the discussion space of the given topic and produces a set of relevant key points (Chapter 3). Second (B), it constructs a model of audience belief and uses it to generate an argument on the topic guided by the constructed discussion space from A (Chapter 4). Finally (C), given the opponent's argument, our approach infers knowledge about the argument's conclusion and weak premises. It then synthesizes a relevant counter that either rebuttal the conclusion or attacks one of the weak premises, guided by the audience model of belief extracted in B and the discussion space from A. Dotted lines in the Figure represent integration steps that we did not address in our experiments.

suitable for a wide range of topics. While these approaches guarantee a degree of synthesizing reliable arguments, they are bounded by the presence of relevant argumentative content in a predefined collection.

On the other hand, generation-based approaches like the work of Hidey and McKeown (2019) consider only a single claim as an input, which is not comprehensive since arguments in real natural texts might contain more than a single claim. The work of Hua and Wang (2018) uses an end-to-end neural network that con-

sumes the input as a sequence of tokens without considering the argumentative relation between premises and their conclusion. In daily-life debates, however, people often do not explicitly state their argument's main point (i.e., its conclusion) since it is often clear from the context (Habernal and Gurevych, 2015) or can be inferred from commonsense knowledge. Consequently, we argue that generating a proper counter becomes more challenging and requires more sophisticated approaches to analyze the input argument. The following section will highlight our contributions towards analyzing different aspects of the opponent's argument and use this to produce more effective counter-arguments.

1.4 Thesis Approach

While the purpose of a debate can vary based on its settings, we consider *maximizing consensuses* as its primary goal, which is different from the goal of winning a debate set by Project Debater (Slonim et al., 2021) in their showcase of debate technologies. Technology can then play an essential role in achieving this goal (as an assistant tool or by autonomously taking part in the discussion) by collecting a broad spectrum of relevant critical points to the discussed topic, challenging the opponent's argument, and focusing on its users' belief system. Therefore, our main research question is: *how to achieve more effective engagement of debate technologies?* Based on the limitations we discussed in the previous section, we devise a set of specific research questions that can be studied independently and design approaches and experiments to evaluate these questions. We summarize these research questions as follows:

- **Modeling Discussion Space:** How to effectively extract relevant key points for a topic that form its discussion space?
- **Modeling Audience's Beliefs:** What models can be used to represent the audience's beliefs and how to use them to generate more effective arguments?
- **Modeling Opponent's Argument:** How to model the opponent's argument and use this knowledge to generate more relevant counters?

As shown in Figure 1.3, our overall conceptual approach consists of three main steps that all together aim to optimize the generation of effective (counter) arguments in debates. The first step (A) starts with a controversial topic and constructs a discussion space composed of key points that one can use in the other components to guide the generated arguments. The second step (B) takes an audience's representation as input. It builds a model of their belief to generate an argument on the topic tuned toward the audience's interest, guided by the discussion space. Finally, the third step takes the opponent's argument and infers its conclusion and its weak premises. It then generates a counter that either rebuttal the conclusion

or attacks one of the weak premises, guided again by the discussion space and the audience model of belief. We provide methods and realizations of most of these steps in our approach, but we leave some integration steps, highlighted in dotted lines in the figure, for future research. The following subsections will provide a detailed overview of our approach's designed components and main findings.

1.4.1 Identifying Discussion Space

As mentioned in Section 1.3, generating relevant arguments requires exploring potential relevant discussion points to the given topic. While retrieving an enormous amount of natural language arguments on a given topic is achievable through argument search engines (Wachsmuth et al., 2017b), distilling them into a set of key points is an understudied task. Hence, we aim to model key point extraction from a given collection of arguments on a topic. Following Bar-Haim et al. (2020b), we define our discussion space as a set of concise statements representing the core argumentation relevant to the given topic – called key points hereafter. We consider each sentence in the collection to be a candidate key point and propose an approach of two steps to extract these key points. The first step models the key point candidacy of each sentence in the argument collection, and the second step ranks and aggregates these sentences according to their candidacy scores into the final set of key points representing the discussion space and diverse enough to cover most aspects mentioned in the collection.

In particular, first, our approach's key point candidacy component generates candidacy scores for each sentence based on their argumentativeness and centrality in the argument collection. We realize this component as a graph-based model that encodes all sentences in the collection as nodes in the graph with edges drawn based on the semantic similarity between sentences. It then runs a variant of PageRank algorithm (Page et al., 1999) biased toward the argumentative scores of these sentences. Second, we realize the key points aggregation component as an algorithm that first ranks sentences according to their candidacy scores and then selects the top k sentences that are diverse enough to form the final discussion space. In a follow-up study, we propose an alternative method to ensure the diversity of selected key points by encoding an extra term reflecting how unique the sentence is to its own argument compared to the whole collection, and we call this term contrastiveness.

In our experiments, we first assess the first component of our approach that models the candidacy of key points on *argument snippet generation* as an intermediary task. Specifically, we consider useful argument snippets to be good candidate key points. Thus, we start by defining the argument snippet generation task as follows:

Given a collection of relevant arguments to a specific topic, extract two sentences from each that represent the corresponding main claim and reasoning behind it.

We then conduct experiments to evaluate our key point candidacy model on this task by selecting the two top-scored sentences as a snippet. Our results demonstrate that our approach outperforms strong baselines such as LexRank (Erkan and Radev, 2004) and BertSum (Liu, 2019) in selecting the best representative sentences. The former baseline also relies on the PageRank algorithm but without the bias towards argumentativeness, and the second baseline is a transformer-based model (Vaswani et al., 2017) fine-tuned towards generating summaries of texts. In the context of argument search, we compare our approach to a query-based snippet generation approach of (Białecki et al., 2012) that extracts sentences from the argument overlapping with the query. We show that our approach produces better representative snippets than this baseline in most cases. Our follow-up study empirically shows that including contrastiveness leads to a trade-off with the representativeness criterion. Biasing extracted key points towards contrastiveness might lead to a loss in how representative they are of their argument and vice versa. In the automatic evaluation, we demonstrate how to balance this trade-off and ensure the extraction of diverse snippets. Our findings support the applicability and importance of considering argumentative, representative, and contrastive sentences to generate useful snippets within the argument search. Hence, our approaches can also produce useful candidate key points. Second, we study the overall task of extracting and aggregating sentences into key points. Due to the similarities between our task and the key point analysis shared task introduced by Friedman et al. (2021b), we assessed the effectiveness of our approach (Alshomary et al., 2021b) by participating in their shared task held at the 8th Workshop on Argument Mining (ArgMining 2021) during EMNLP 2021. In the manual evaluation run by the organizers, our approach ranked best among the submitted approaches.

To summarize, modeling the argumentativeness and contrastiveness of a sentence, along with its representativeness, is essential to extracting useful key points that can form the discussion space of a topic. In the context of argument search engines, the user's search goal is to understand the main claim and reasoning behind a retrieved argument. Therefore, we also found that general-purpose snippets are insufficient for this task. Finally, our main contributions are the following:

- We propose approaches to model the key point candidacy of sentences and to aggregate them into an overall discussion space.
- In the context of argument search engines, we define the task of argument snippet generation and provide a dataset to study this task.
- We provide empirical evidence to support the effectiveness of our approach to model key points.

Remarks Finally, in addressing the task of identifying discussion space, we restricted our view to extractive approaches since they are more intuitive, reliable, and equally appreciated compared to abstractive approaches (Chen et al., 2020). Nevertheless, in doing so, we might fail to address cases where the reasoning or their implications are left implicit in the argument. In such scenarios, abstractive approaches infer these implicit components in an argument.

In the context of debates, human contenders might use an argument search engine (Wachsmuth et al., 2017b) as a source of information to consume potential arguments on the web. Efficient presentation of retrieved arguments in such search engines is crucial to boost their usability. Our proposed approach to generating *snippets* for arguments is one way of boosting this efficiency because it helps users assess the relevancy of the retrieved arguments to their information needs.

1.4.2 Modeling Audience's Beliefs

Despite the apparent importance of audience in assessing argument effectiveness, as mentioned in Section 1.3, research on argument generation so far did not consider generating argumentative texts subject to a given audience. Thus, our goal here is to cover this research gap. On this account, we detail our research question on modeling users' beliefs as follows: (1) How to computationally represent an audience's beliefs and use them to guide argument generation? and (2) Do arguments that focus on the audience's beliefs have more effect on the audience? To study these research questions, we start by formulating the task of belief-based argument generation as follows (Alshomary et al., 2021a):

Given a controversial topic and a representation of the audience's beliefs, generate argumentative text that is both relevant to the topic and matches the beliefs.

As illustrated in Figure 1.3, our approach to this task has two components. The *modeling audience* component takes as an input representation of the targeted audience. It then infers a belief model that guides the argument generation component in generating a final argument that is relevant to the topic and targets the given audience. In our experiments, we consider two representations of the audience's beliefs, their stances on known controversial issues, and their morals based on the moral foundation theory (Haidt, 2012). We then introduce two realizations of our approach: once using an underlying neural generation model (Alshomary et al., 2021a) and once using a retrieval-based model (Alshomary et al., 2022a).

In particular, for stances on controversial topics as a representation, we propose two models. One builds on Li et al. (2016), equipping a neural model with a context vector representing the given stances. The other realization infers a vocabulary from the targeted user's stances on big issues representing their beliefs. It then uses this vocabulary to control the output of a pre-trained neural model using the algorithm of Dathathri et al. (2020) to generate argumentative texts resembling the targeted user's beliefs. We evaluate these models empirically on the *debate.org*

dataset of Durmus and Cardie (2018), which contains users' arguments on various controversial topics and their stances towards the most popular ones on the website (named big issues henceforth). In our experiments, we compare both models against their unconditioned correspondents (i.e., the same models without knowledge about a user), assessing the similarity of generated claims to the ground truth and the likelihood of carrying textual features that reflect users' stances on big issues. Our results demonstrate the ability of our models to encode users' beliefs represented as stances on big issues into generated claims, with the pre-trained language model generating more coherent claims due to the pre-training process.

The second representation we consider is the moral foundations, following our previous work (Alshomary et al., 2022a). Here, we study the feasibility of generating morally framed arguments computationally and their effect on different audiences. Since we aim to evaluate argument effectiveness on human audiences, the quality of these arguments is crucial. Therefore, we ensure this quality by relying on Project Debater (Slonim et al., 2021), a retrieval-based approach of multiple components designed to generate arguments of high quality that compete with human arguments. One of the main steps in our retrieval-based approach is to classify argumentative sentences based on their moral focus, which requires training a classifier for this task. To develop such a classifier, we rely on distant supervision: We use the Reddit dataset of Schiller et al. (2020), which contains argumentative texts with annotated aspects, along with the moral-to-concept lexicon of Hulpus et al. (2020) for the automatic mapping from aspects to morals. Then, we train a transformer-based neural network (Vaswani et al., 2017) on this dataset, achieving high effectiveness on the moral dataset of Kobbe et al. (2020) compared to ablation baselines. To assess the effect of morally framed arguments on a particular audience, we consider liberals and conservatives as alternative audiences. We designed a user study where we separately asked three liberals and three conservatives to rank different arguments on specific controversial issues based on their effectiveness in challenging or empowering their stances. The results suggest that both liberals and conservatives value morally framed arguments more than the general ones. We also found that liberals value arguments that focus on their own morals (care and fairness) the most. At the same time, conservatives, when their stance is challenged, tend to rate arguments focusing on loyalty, authority, and purity higher than the generic ones.

To summarize, we found that it is computationally possible to infer a model of an audience's beliefs based on their stances on known issues or their morals and use these models to tune generated arguments accordingly. Our experiments demonstrate that arguments focusing on morals are more effective than their general counterparts. Also, differently framed moral arguments have varying effects on audiences based on the audience's belief system. Our main contributions towards modeling the audience in argument generation are the following:

- We introduce the belief-based claim generation task
- We propose two approaches to model and encode users' beliefs in the process of argument generation.
- We provide empirical evidence of the importance of considering the audience's beliefs when synthesizing arguments.

Remarks We considered the audience to be a third party in the debate. However, one can also consider the opponent as an audience. Under this condition, our approach can then learn from the opponent's arguments a model representing the human values that concern the opponent and build its argumentation on this basis.

Nevertheless, we acknowledge that the task of tuning arguments toward a specific audience raises some ethical concerns. One might consider this as manipulation. However, changing others' minds is considered manipulation only when done deceptively. Therefore, transparency is a key aspect in any approach for this task, where users have the right to be informed about any potential employment of their beliefs in generated arguments. This also requires models to be interpretable, enabling a good understanding of their workings. Moreover, while relying on the moral foundation theory to model an audience is a good start, one could point out the inherent limitation of this approach since it reduces the human condition into only five concerning moral foundations.

1.4.3 Countering the Opponent's Argument

In debates, as highlighted in Figure1.2, an important step is to synthesize a proper counter-argument to the opponent's argument. According to Walton (2009), a model for countering an argument requires knowledge about the argumentation structure of the given argument to be able to rebut, undercut, or undermine it.

In this work, our core research question is how to analyze the opponent's argument and use this knowledge to generate relevant counters. Hence, we investigate different methods of extracting and infusing such knowledge into neural counter-argument models. In particular, we explore two types of knowledge: (1) weak premises in the input argument and (2) the inferred argument conclusion. For this purpose, we study the task of *conclusion inference* and the *identification of weak premises* to use them in the counter-argument generation task by modeling it as an argument undermining or rebuttal. As illustrated in Figure 1.3, our overall approach then takes an argument in natural language text as an input, identifies its conclusion and weak premises, and then uses this knowledge to generate a relevant counter-argument that either rebuts or undermines the input argument.

In particular, we first look at how to infer a conclusion from a set of premises (Alshomary et al., 2020b). We define an *argument conclusion* as a statement carrying a stance towards a specific target phrase. For example, given the claim *Human*

life is invaluable, the target is the phrase *Human life*, and the claim holds a *pro* stance towards it. Hence, we decompose the conclusion inference task into three main steps: (1) inferring the conclusion's target from the premises, (2) inferring the conclusion stance, and (3) generating the conclusion's text with the inferred target and stance. To this end, we focus on the conclusion target inference step due to the limitation of text generation models and the absence of big corpora to study this task. We hypothesize that the conclusion target is related to the targets of the argument's premises. Therefore, our approach identifies these premise targets and then learns via a triplet neural network (Hoffer and Ailon, 2015) to infer a proper embedding representation of the conclusion target. We then use this embedding to select a conclusion target from a predefined knowledge base of concepts. Our experiments show empirical evidence of premise targets' importance in the conclusion generation process. Later, with the success of pre-trained language models, we also experimented with their effectiveness on the task by jointly modeling the conclusion and counter-argument generation tasks.

As mentioned, one can identify a weak point/premise and use it to undermine the argument. To identify the attackability of a premise in an argument, similar to the work of Jo et al. (2020), we learn this criterion from data, namely the *Change My View* Subreddit (CMV). CMV is a debate forum where commentators attack an original post by quoting one of its supporting premises, signaling a potential weakness. We hypothesize that the attackability of a premise is better learned by considering both the conclusion and other premises of the argument (Alshomary et al., 2021c). Thus, unlike previous work, which models each premise's attackability independently, we model it as a ranking task, where we learn to rank all argument's premises jointly with respect to their conclusion (Alshomary et al., 2021c). Our automatic evaluation demonstrates a significant gain of effectiveness on this task compared to previous related work.

Given that we can infer the argument's conclusion and identify its premises' attackability, we turn our view to how we can use this knowledge to generate more effective counter-arguments. For this, we propose two alternative approaches that approximate argument undermining and rebuttal phenomena. Both approaches generate text from scratch, utilizing the power of pre-trained language models. For argument undermining, our approach uses the knowledge learned about weak premises to generate a counter that attacks the top k weak premises. We encode this knowledge as extra information on the token-level, reflecting whether the token belongs to a weak premise. The model then uses this knowledge to learn the best counter-argument from the data. In our experiments, we compare this approach against the same architecture but trained without information about the weak premises. Our results highlight the gain of inducing knowledge about premise attackability to the counter-argument generation task.

We further investigate whether inferring the argument's conclusion can improve the effectiveness of counter-argument generation task. We propose to model the two tasks jointly through a transformer-based model. Additionally, in the inference time, generating the conclusion allows us to assess the suitability of the generated counter in terms of its stance towards the inferred conclusion. Hence, we add to our model a stance-based component that ranks a set of candidate counters based on their stance toward the inferred conclusion and selects the top one as the final counter. We conduct an experiment to compare our model's effectiveness against other baselines. Results show that inferring the conclusion leads to more relevant counters and generates more stance-accurate ones.

To summarize, we found that conclusions only sometimes explicitly appear in argumentation, and to infer them accurately, one can model the relation between the premise and conclusion target. Moreover, we discovered that weak premises are better identified jointly with respect to the argument conclusion. Finally, we found that inferring the argument conclusion and its weak premises can lead to synthesizing more effective counters. We summarize our contributions as follows:

- We propose an approach to computationally model argument undermining that identifies weak premises in an argument and then generates a corresponding counter.
- We propose a multitask approach that generates the conclusion and a corresponding counter to rebut the argument.
- We provide empirical evidence for the importance of identifying the argument conclusion and its weak premises as part of the process of counter argument generation.

Remarks In the context of knowledge-enhanced text generation (Yu et al., 2022), so far, our approaches exploited *internal knowledge* from the given input argument. Nevertheless, other types of knowledge, like commonsense knowledge, can also be used to boost counter-argument generation models. Such an approach can infer the main targeted concept in the input argument and the stance towards it. The model then uses a commonsense knowledge base (causal relations) to output a proper counter-argument. Moreover, since we deal with text generation, we have inherent challenges, such as ensuring the factual correctness of generated texts. Research on claim verification enjoyed a lot of attention in recent years. One can utilize a claim verification component in their argument generation model to self-check the generated arguments to address the factual correctness of generated arguments.

1.5 Thesis Structure

In the following chapters, we will first start by providing the necessary background and related work (Chapter 2), which covers topics like the basic blocks of natu-

Publication	Venue	Topic	Chapter
Alshomary et al. (2020b)	ACL	Conclusion Inference	Chapter 5
Alshomary et al. (2020a)	SIGIR	Modeling Key Points	Chapter 3
Alshomary et al. (2021a)	EACL	Stance-based Modeling	Chapter 4
Alshomary et al. (2021c)	Findings of ACL	Argument Undermining	Chapter 5
Alshomary and Wachsmuth (2021)	Patterns Journal	Audience-based Argument Generation	Chapter 4
Alshomary et al. (2021b)	ArgMining	Key Points Generation	Chapter 3
Alshomary et al. (2022a)	ACL	Moral-based Modeling	Chapter 5
Alshomary et al. (2022b)	COMMA	Modeling Key Points	Chapter 3
Alshomary and Wachsmuth (2023)	EACL	Argument Rebuttal	Chapter 5

TABLE 1.1: List of peer-reviewed papers throughout this dissertation work including the venue where the paper is published, the topic it covers, and where it appears in this dissertation

ral language processing methods (Section 2.1), machine learning techniques used for computational argumentation. Following that, Chapters 3, 4, and 5 will cover our research's three main contributions, which also resonate with the three main tasks a debate technology should address. First, in Chapter 3, we discuss the first task of discussion space identification of the given input topic. We first present our approach to this task that contains two main components; *modeling key point candidacy* and *key point aggregation*. Section 3.1 presents our two implementations of modeling the key point candidacy, the Representativeness and Contrastiveness approaches. We then present a series of experiments in Section 3.3, including the argument snippet generation study (Subsection 3.3.1) and the overall evaluation of our approach to the task of key point analysis (Subsection 3.3.2).

In Chapter 4, we turn our view to the second contribution that addresses the audience in argument generation. We introduce the task of audience-based argument generation, and our approach consists of two main components, *modeling audience* and *argument generation*. Section 4.1 then discusses the two audience representations we considered and how we computationally model them. Section 4.2 presents how we use these two audience models to adjust the generated arguments to fit the corresponding audience. We finally present the experiments conducted to evaluate our proposed approach in Section 4.3.

Chapter 5 will then present the third contribution focusing on countering the opponent's argument. We argue that specific knowledge about the argument structure is needed to generate successful counters; the argument conclusion and its weak premises. We first discuss our approach to this task, which consists of two

Publication	Venue	Topic
Ajjour et al. (2019a)	EMNLP-IJCNLP	Modeling Frames in Arguments
Syed et al. (2021)	Findings of ACL	Generating Informative Conclusions
Kiesel et al. (2022)	ACL	Identifying Human Values behind Arguments

TABLE 1.2: List of relevant peer-reviewed papers that we co-authored with other fellow researchers in the field.

components; *argument analysis* and *counter generation*. In Section 5.1, we present two tasks we address in the argument analysis component. First, we study the conclusion inference and propose an approach to this task (Subsection 5.1.1), and then the weak premise identification task along with our proposed approach (Subsection 5.1.2). Afterwards, we move our view to the counter generation component. We first present our argument undermining implementation (Subsection 5.2.1) that uses the weak premises to generate a counter, then present the argument rebuttal approach (Subsection 5.2.2), which models the conclusion and counter generation jointly. Finally, Section 5.3 presents a set of experiments we conduct to evaluate each of the proposed approaches.

We finally conclude our thesis work in Chapter 6 with implications of our work, key findings, and future outlooks. These three main contributions can be integrated into the big picture of a debate technology that can either engage in a debate or assess a human debater to boost its effectiveness in argument generation to maximize agreements in debates.

1.6 List of Publications

This thesis is based on a set of peer-reviewed scientific papers we published at international conferences in the fields of computational linguistics, information retrieval, and computational argumentation. In comparison, these publications propose approaches to specific tasks, while our thesis stitches together these approaches by providing an overview of how each presented approach can be integrated to reach an overall framework for a debate technology that is more effective in its engagement in debates. Additionally, in this thesis, we reflect on the societal impact of our contributions and provide a future outlook on how the research in this field might progress. Table 1.1 gives an overview of each of the published works along with the venue where it is published, the topic it covers, and where it appears in this thesis. Additionally, Table 1.2 lists other publications that we co-authored with other researchers from the field that cover relevant topics such as identifying frames in arguments, using language models to generate informative conclusions, and developing classifiers to identify human values in argumentative texts.

Chapter 2

Background and Related Work

At its heart, our work researches methods to analyze and synthesize argumentation in natural language texts to enable an effective engagement of technologies in human debates. That said, this thesis lies in the intersection between the Natural Language Processing (NLP) and the Argumentation fields. On the one hand, our research is informed by argumentation theories that model the composition of an argument and its interaction with other arguments, as well as the use of argumentation for communication. On the other hand, we study the argumentation phenomena in natural language texts, so our approaches build on and contribute to the NLP field. Moreover, argumentation is a social activity where humans engage in discussions on controversial issues to achieve agreement on a decision or to convince each other. Hence our research is also inspired by social science theories that address the human condition in argumentation, such as belief systems and their relation with argument effectiveness. Lastly, inherited from the NLP field, our research is empirical, using Machine Learning (ML) and Artificial Intelligence (AI) methods to learn a model of argumentation from natural language texts.

In the following, we will start in Section 2.1 by covering the foundational concepts of NLP, such as how to represent texts to ML algorithms, what evaluation measures are used to assess algorithms' effectiveness and popular machine learning algorithms that are usually used to address NLP tasks. Next, Section 2.2 will introduce the field of argumentation, discussing argumentation frameworks that model argument structures and relations and the role of the audience in these frameworks. Moreover, we will provide a short overview of the social science perspective on argumentation and explain the moral-foundation theory on which we base some of our work. Section 2.3 will cover the various tasks, models, and applications of argumentation in the field of natural language processing, which form the related work to our thesis.

2.1 Natural Language Processing

Natural language processing (NLP) is a linguistic-informed field that uses machine learning techniques to address tasks that enable computers to process human language. Such tasks are text translation (Tan et al., 2020), document summarization (El-Kassas et al., 2021), and sentiment analysis (Wankhade et al., 2022). Research in NLP is carried out empirically, where the effectiveness and efficiency of developed approaches are tested experimentally on a sample dataset of texts. The NLP field has witnessed two transitions in the popular techniques used to address NLP tasks. The first transition happened with the adaption of Machine Learning (ML) methods to learn the solution for a corresponding task from data. Before that, methods were mainly rule-based, where expert knowledge is encoded as a set of algorithmic steps and rules to extract information from natural language texts (Taboada et al., 2011). Until recently, developing approaches to solve NLP tasks required task-specific hand-crafted features to represent the input texts for an ML model. However, the second transition in the field came with the rise of pre-trained language models (PLM), which are deep neural networks that are trained to model human language on big corpora and able to transfer this knowledge to solve various downstream tasks (Radford et al., 2019). These PLMs significantly boosted state-of-the-art results on all NLP tasks but also raised concerns about their decisions' interpretability, bias, and fairness. In this section, we will first provide an overview of NLP tasks, how the text gets processed and represented to an ML algorithm, and the evaluation metrics typically used to assess the effectiveness of NLP approaches. Second, we will focus on ML-based methods for solving NLP tasks, where we discuss different training paradigms and the popular neural network architectures (NN) used in this field.

2.1.1 NLP Tasks and their Evaluation

Natural Language Processing tasks can be categorized into two main areas: text analysis and generation. In both areas, input texts are represented as a sequence of tokens or characters and mapped into some vector space on which ML algorithms can operate. The output of text analysis tasks is information that can be on different levels of granularity. For example, on the token level, one might be interested in classifying the part of speech of each token (verb, noun, etc.) or whether a token is a discourse marker (because, therefore, etc.). On a higher level, the task can be to categorize the stance of the whole text (sequence of tokens) toward a specific topic. In text generation, the output is a new sequence of tokens, for example, a translation of the input text or a conclusion of the input argument. In the following, we will describe how texts get represented by machine learning algorithms and how approaches to these tasks are evaluated.

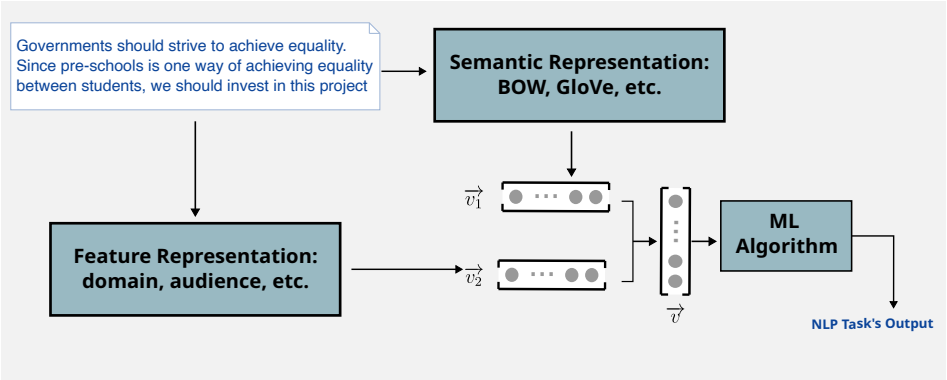

FIGURE 2.1: The input to the ML algorithm is a vector representation of the text that concatenates two representations. The first is the semantic representation of the text through models like BOW or GloVe, while the other representation can be a set of other task-specific features that models certain contextual information about the text, such as the source domain, targeted audience, etc.

Text Representation

Typically, as illustrated in Figure 2.1, most NLP methods process text by first splitting it into a sequence of tokens and then mapping it into a single feature vector $\overrightarrow{v}$ aggregated from its tokens' semantic representations $\overrightarrow{v_1}$ and potentially another task-specific feature vector $\overrightarrow{v_2}$ such as the source domain, targeted audience, etc (Jurafsky and Martin, 2000). As for the semantic representation, traditional methods like Bag-of-Words (BOW) and the Term Frequency–Inverse Document Frequency (TF-IDF) take a predefined vocabulary (a dictionary of tokens) of size N and map the text into one vector of N dimension with a value assigned to each index reflecting the corresponding token's weight in the text (Salton and Buckley, 1988). For example, in TF-IDF representation, this weight can be computed from the token's frequency in the text weighted by its importance in the vocabulary.

However, these methods create sparse representations, making it hard for ML to efficiently learn meaningful representations of texts (Bengio et al., 2000). Therefore, recent methods have been proposed to learn dense token representations from big text corpora by either exploiting corpus statistics (Collobert, 2014) or modeling n-grams (sequences of n tokens) across an entire corpus (Mikolov et al., 2013). These methods, such as GloVe embedding (Pennington et al., 2014) and CBOW (Mikolov et al., 2013), became the basis of text representation in most NLP tasks. They typically learn a dictionary that maps a predefined set of tokens into their semantic vector representation. An ML algorithm can then use this dictionary to project a text (sequence of tokens) into a single semantic vector by averaging all its tokens' vectors, for example.

Using GloVe-like embeddings, each token in a vocabulary is associated with a single semantic vector regardless of the surrounding tokens. Nevertheless, these static methods do not capture the change in the word meaning based on the context. Therefore, a new class of context-sensitive embedding methods has been proposed to address this limitation. The primary component of these methods is a neural network (NN) trained on a huge corpus of unlabeled texts to model human language by predicting the next token given a set of previous or surrounding tokens. It has been found that the hidden states of these trained models are good context-sensitive representations of the input tokens (Peters et al., 2018). These pre-trained NNs, such as the BERT model (Devlin et al., 2018), can then be used in downstream NLP tasks to extract dynamic semantic representations from the input text. Furthermore, starting from a general semantic embedding space, one can further learn a more task-specific embedding space by utilizing a contrastive learning paradigm, as we will see in Subsection 2.1.2. In this thesis, we use the CBOW model Mikolov et al. (2013) to represent the conclusion and premise targets in the embedding space, and the Sentence-BERT model (Reimers and Gurevych, 2019), a tuned version of BERT towards modeling sentence similarity, to represent arguments and their corresponding key points in the embedding space.

Evaluation Methods

Approaches to address the various NLP tasks are empirically evaluated on a corresponding corpus of texts that is typically split into training, validation, and test splits. During the development stage, the proposed approach is trained, if needed, on the training split and evaluated on the development dataset iteratively to find the best hyper-parameters. The final developed version of the approach is then tested on the test set to compare against other baseline approaches to the task. For text analysis tasks, evaluation measures such as *F*1-*score* and *accuracy* are typically used to assess the model's effectiveness automatically. While accuracy quantifies this effectiveness as the number of correct classification decisions across all classes, the F1-score computes a class-specific score that is a harmonic average of the model's precision and recall. In particular, to compute a model's F1-score for a specific class c, one first computes the number of true positives tp (instances of class c that were correctly classified as such by the model), false positives fp (instances of other classes that are classified as c), false negatives fn (instances of class c that are classified as other classes by the model), and the true negatives tn (instances that are correctly not classified by the model as c). Then compute the precision and recall according to Equation 2.2 and 2.3. Finally, the F1-score can be computed as the harmonic mean between precision and recall.

$$\text{F1-score} = \frac{2 \cdot (\text{Precision} \cdot \text{Recall})}{(\text{Precision} + \text{Recall})} \tag{2.1}$$

$$\text{Precision} = \frac{tp}{tp + fp} \tag{2.2}$$

$$\text{Recall} = \frac{tp}{tp + fn} \tag{2.3}$$

However, evaluating text generation approaches is trickier. Commonly, given a test dataset with its ground-truth texts, a model's effectiveness can be assessed by computing the lexical overlap between the generated text and the ground-truth one. This kind of assessment forms the basis of most text generation evaluation measures such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and METEOR (Banerjee and Lavie, 2005). For example, as shown in Equation 2.4, to compute the BLEU score between a predicted text t and a reference r, one first computes the model's precision at different n-grams levels (p_n) as the fraction of predicted n-grams that appear in the ground-truth text for $n \in [1 \cdots N]$. Then the geometric weighted average precision is computed and multiplied by the brevity penalty BP (Equation 2.5) that penalizes short sentences.

$$\text{BLEU} = \text{BP} \times \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{2.4}$$

$$\text{BP} = \begin{cases} 1 & \text{if } t > r \\ \exp^{(1-\frac{|r|}{|t|})} & \text{otherwise} \end{cases} \tag{2.5}$$

Other measures, such as the ROUGE score, quantify text similarity in terms of precision and recall, while the METEOR metric works by computing the mapping of words and synonyms between the generated and ground-truth texts and calculating the F-score based on these mappings. Another class of text generation measures is embedding-based, such as BERT score (Zhang et al., 2019). Under this category, the comparison is carried out between the embedding vectors of the generated t and the ground-truth r texts using metrics like cosine similarity (Equation 2.6) to compute the semantic rather than the lexical similarity between texts.

$$\text{cosine} = \frac{\overrightarrow{r} \cdot \overrightarrow{t}}{\|\overrightarrow{r}\| \|\overrightarrow{t}\|} \tag{2.6}$$

Nevertheless, these methods fall short on tasks where multiple generated texts can be correct, and there is no single ground truth. For example, multiple valid arguments might exist on a given topic in computational argumentation tasks. There-

fore, besides automatic evaluations using the mentioned measures, a manual evaluation of generated texts is usually done via a user study. In these user studies, a group of human users is asked to score different criteria of the generated texts. For example, to assess a generated argument, one might ask whether the argument is informative, argumentative, or grammatically correct. The scores of single human evaluators are then aggregated per sample text by either taking the average or the majority score. Further, an inter-annotator agreement score is computed to quantify the degree of agreement between the annotators. For example, given m human judging N instances by assigning one of k categories for each instance, the agreement according to Fleiss' κ (Fleiss, 1971) is computed from two main components; the observed agreement p_o and the chance agreement p_e. As shown in Equation 2.9, the chance agreement p_e is computed from the proportion of all assignments P_j for each class. The observed agreement p_o is then computed by first calculating the proportion of annotator pairs in agreement out of all annotators P_i for each instance and then aggregating the scores as shown in Equation 2.8. Finally the Fleiss's κ is computed according to Equation 2.7.

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{2.7}$$

$$P_o = \frac{1}{n} \sum_{i=1}^{N} P_i \tag{2.8}$$

$$P_e = \frac{1}{n} \sum_{j=1}^{k} P_j^2 \tag{2.9}$$

Typically, this agreement score gives insights into the difficulty or subjectivity of the questions asked in the evaluation task and the degree of reliability with the study results to be trusted.

2.1.2 Learning Paradigms

To learn ML models from data, we can employ several learning schemes for different task settings. The most popular two schemes are the **supervised** and **unsupervised** learning. In the former, we start from a dataset of input texts with corresponding labels of interest. For example, an input text of a tweet and its sentiment label being positive or negative. The model is then trained on this data to learn a mapping function between the text represented via features (e.g., embedding) and the sentiment labels. Unsupervised learning, however, tries to find patterns in the data without the need for ground-truth labels. Clustering arguments per topic is an example of a task that can be addressed through unsupervised learning. Besides

these two training schemes, in this thesis, we employ **multitask learning** and **contrastive learning** (Chopra et al., 2005) as other means of training models on data.

In Multitask learning, a single model is optimized to jointly solve two or more different tasks at once, for example, to classify the sentiment of a tweet and whether it is misinformation. This scheme allows the model to learn new patterns that generalize beyond each task. The model sharing between the two tasks can be generally classified into hard parameter or soft parameter sharing. In the former, the same model parameters are used to optimize the two tasks, while in the latter, two different sets of parameters are utilized. However, regularization is applied during training to reduce the differences between these two sets of parameters. Zhang et al. (2022) provide a comprehensive survey of the recent trends in multitask learning. Our work uses hard parameter sharing between the conclusion and counter-argument generation tasks in a multitask setting. As shown in Equation 2.10, the total loss of the model L is a weighted sum of the two losses. The parameters λ_1 and λ_2 are either predefined or dynamically learned during training (Kongyoung et al., 2020).

$$L = \lambda_1 \cdot loss_1 + \lambda_2 \cdot loss_2 \tag{2.10}$$

In the contrastive learning paradigm, we aim to learn a distance-based function that maps instances to a new latent space where similar instances are closer than dissimilar ones. In particular, given a set of pairs of instances (x_1, x_2) with the corresponding binary labels y reflecting whether they are similar ($y = 1$) or not ($y = 0$), they are first encoded using an embedding function (e.g., neural network) into $(\overrightarrow{x}_1, \overrightarrow{x}_2)$. Then, as illustrated in Equation 2.11, a distance-based score D (e.g., cosine distance) is computed. The objective function then is a sum of two parts. The first part is minimized for similar instances ($y = 1$), while the second is maximized for dissimilar instances of label $y = 0$.

$$L = y \cdot D(\overrightarrow{x}_1, \overrightarrow{x}_2) + (1 - y) \cdot max(0, m - D(\overrightarrow{x}_1, \overrightarrow{x}_2)) \tag{2.11}$$

m is a hyper-parameter representing the lower bound distance between dissimilar pairs. Our thesis uses contrastive learning to learn a similarity function to match extracted key points to arguments. Additionally, we utilize a triplet loss function based on the same idea of contrastive learning to match premise targets to their corresponding conclusion target. The triplet loss function takes as an input a triplet of an anchor a, a positive x_1, and a negative x_2 instance and learns to minimize the distance between the embeddings of the anchor and the positive pairs while maximizing it for the anchor and negative embeddings:

$$L = max(0, D(\overrightarrow{a}, \overrightarrow{x}_1) - D(\overrightarrow{a}, \overrightarrow{x}_2) + m) \tag{2.12}$$

In this thesis, most of our approaches are supervised learning on relevant datasets collected for our tasks. Nevertheless, we also rely on contrastive learning to model the semantic relation between the conclusion and premise targets in the embedding space (Section 5.1.1). Additionally, we employ a multitask learning paradigm to jointly learn to predict the conclusion and the counter of an input argument (Section 5.2.2).

2.1.3 Deep Neural Networks

While several neural frameworks can be employed to solve NLP tasks, the most prominent ones are recurrent neural networks (RNNs) and transformer architecture. **RNNs** process texts sequentially, taking a single token as input at each time, enabling the processing of texts of different lengths (Salehinejad et al., 2017). They are equipped with a hidden state vector that maintains a representation of the so-far processed tokens. At each time step, the previous hidden state and the token representation are used to compute the model's current state. The final updated hidden state after consuming the last token carries then the semantic representation of the whole input text on which a task-specific classification layer can be applied. Different variations have been proposed, such as LSTMs (Graves et al., 2013) and GRUs (Cho et al., 2014), to address the RNNs shortcomings, such as the vanishing gradient problem.

The **transformer** architecture was then proposed by Vaswani et al. (2017) to replace RNNs with an architecture based only on the attention mechanism. As illustrated in Figure 2.2, transformers are composed of an encoder and a decoder, each consisting of a stack of fully connected layers. Each layer in the encoder consists of two sub-layers, one that utilizes a *multi-head self-attention mechanism*, and the other is a simple feed-forward network. As highlighted in Equation 2.13 and 2.14, the self-attention mechanism contextualizes each token along with all other tokens in the text. Specifically, first, the input embeddings X are passed through the query (Q_θ), key (K_θ), and value (V_θ) weight matrices, then a dot-product is computed between Q and K for all tokens in the input sequence. The output of the attention layer is finally a weighted sum of the V vectors, where the attention scores determine the weights. In the decoder layers, the same self-attention mechanism is used, but it also accounts for the output of the encoder.

$$Q = XQ_\theta, K = XK_\theta, V = XV_\theta, \tag{2.13}$$

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2.14}$$

Nevertheless, different variations of transformer-based architectures can be implemented via only the encoder layers (Radford et al., 2018). Unlike RNNs,

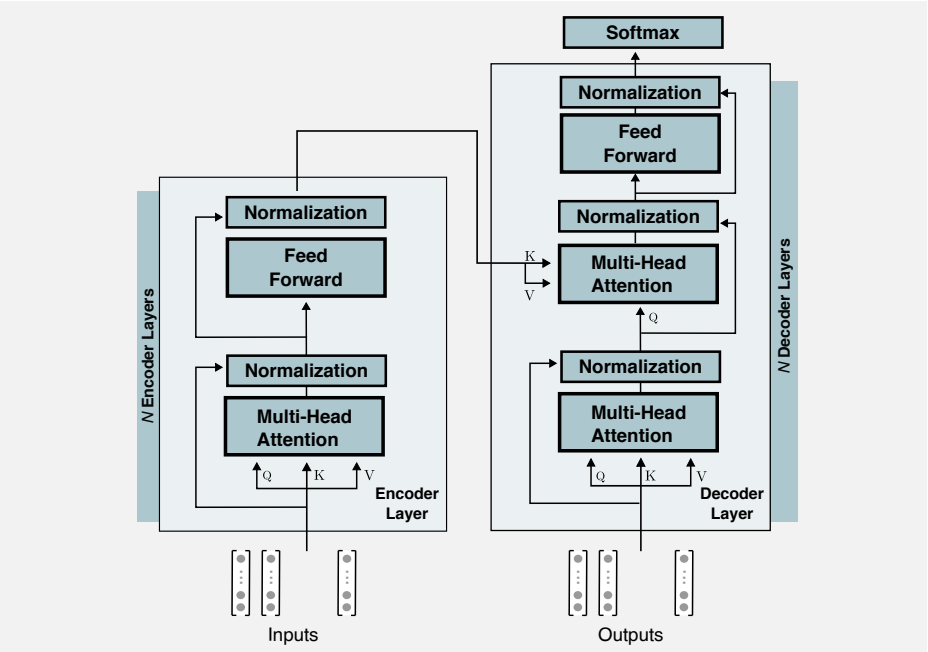

FIGURE 2.2: The transformer-based neural network is an encoder-decoder architecture of N encoder or decoder layers. The encoder layer takes a sequence of token embeddings as an input and performs multi-head attention. The output of the multi-head attention component is then passed to a feed-forward layer. The decoder layer performs the same steps on the output sequence, but it consists of an extra multi-head attention component on the output of the encoder.

transformers enabled processing sequences in parallel, boosting training efficiency while also achieving new state-of-the-art results on several NLP tasks. This gain in efficiency allowed the training of large transformer-based models in an unsupervised way on massive corpora leading to the emergence of pre-trained large language models that can be fine-tuned on downstream tasks and achieve state-of-the-art results. Therefore, most approaches in this thesis use different variations of transformer-based architectures. In the following, we will discuss transformer-based LLMs and their usage in NLP.

2.1.4 Pre-trained Language Models (PLM)

As mentioned, transformers enabled a new learning scheme in which models are first *pre-trained* on big corpora in an unsupervised manner, then *fine-tuned* on a downstream task in a supervised way. In the pre-training phase, models are trained on tasks pertaining to modeling human language, like predicting the next token or filling in the blank. Hence, a massive corpus of natural language texts (e.g., the web) can be used for training—the pre-training stage results in a model that can

produce meaningful contextual representations of texts. In the second stage, given an NLP task (e.g., sentiment analysis) and a small to medium size labeled dataset, a pre-trained model can be further trained (e.g., fine-tuned) for only a few epochs to achieve state-of-the-art results.

The most popular LMs in the field are BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018) and GPT (Generative Pre-trained Transformer) (Radford et al., 2018). The former models token jointly by considering both left and right contexts. The model is trained to reconstruct masked sequences of tokens as well as predict whether the following sentence is naturally occurring. In contrast, GPT (and its followers GPT-2 and GPT-3) is an autoregressive model trained on predicting the next token given a sequence of previous tokens. Research has demonstrated powerful abilities that emerge from pre-training large LMs (LLMs) on big corpora. For example, Jawahar et al. (2019) showed that the BERT model could implicitly learn to capture the structure of the language. Radford et al. (2019) highlighted the ability of GPT to solve tasks such as language comprehension and question-answering tasks also implicitly during its unsupervised pre-training phase.

2.1.5 Text Generation

Computational generation of text is among the most challenging tasks in NLP due to the imbalance between the input and output of these tasks, the grammatical complexity, and the semantic ambiguity of natural languages (Lu et al., 2018). In general, Garbacea and Mei (2020) categorized text generation tasks into free-text, controlled, and constrained text generation. This thesis deals with the natural language generation of arguments that can be categorized under the two latter categories. On the one hand, it is a controlled-text generation in that we always have additional information (c) that controls the generated texts in the form of an input sequence representing a topic, an argument, or an audience representation. As shown in Equation 2.15, the task is to learn a distribution $p(y|c)$ of the output sequences conditioned on c.

$$p(y|c) = \prod_{i=1}^{n} p(y_i|y_{y<i}, c) \tag{2.15}$$

On the other hand, in some cases, we also perform constrained text generation by ensuring that the generated argument contains certain concepts or it follows specific rules, like having a certain stance towards a topic.

Models

Typical architectures to address controlled text generation are those that perform sequence-to-sequence mapping, called Sequence-to-sequence (Seq2seq) models.

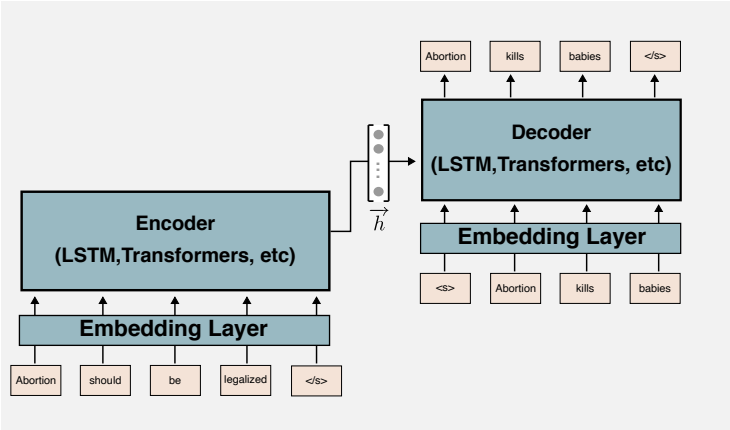

FIGURE 2.3: Sequence-to-sequence architecture consisting of an encoder that consumes the input text and produces one hidden state, and the decoder that consumes the output sequence shifted to the right given the hidden state. Both the encoder and decoder blocks can be implemented by using an LSTM neural network or transformers.

As illustrated in Figure 2.3, they consist of two building blocks, an encoder that consumes the input sequence and projects it into a single latent representation and a decoder that decodes the output sequence one token at a time, given the encoder's output representation. The underlying model for these building blocks can be any appropriate neural network, such as LSTMs, or Transformers. The objective function for these models is then to maximize the conditional log-likelihood of generating the correct output sequence y given the input sequence x and trained on pairs of input/output sequences:

$$-\log L_\theta = -\sum_{i=1}^{|\tau|} \log\ p_\theta(y_i|x_i) \tag{2.16}$$

Where θ is the learned parameters, τ is a set of training pairs of input and output sequences (x and y). To boost their effectiveness, as mentioned in Subsection 2.1.4, typically, one starts from a pre-trained version of these building blocks (encoder/decoder) and fine-tunes it further on the dataset that resembles the task to be learned. A popular pre-trained Seq2seq model is called BART, proposed by Lewis et al. (2020). The BART model consists of a bidirectional transformer-based encoder (similar to BERT) and an autoregressive transformer-based decoder (similar to GPT). It is pre-trained as an autoencoder to reconstruct corrupted input texts. The authors experimented with different techniques of corrupting the input texts, such as shuffling the order of the input sentences or replacing random spans of text with a single mask token. The model achieved state-of-the-art results by fine-tuning it on several text generation tasks. Moreover, controlled text generation can also be performed via autoregressive models such as GPT (Radford et al., 2018)

that consists of only a set of transformer-based decoder layers. In this case, the input sequence is mapped into an embedding. The decoding layers then resume generating the next tokens in an autoregressive manner given the previous tokens.

Our work uses the BART model to learn to generate the conclusion and the corresponding counters for an input argument (Section 5.2.2) and GPT-2 model to generate arguments on a specific topic targeting a particular audience (Section 4.2).

Decoding Strategies

Various decoding strategies exist in the literature to generate texts from trained Seq2seq models. The simplest is the greedy search (Equation 2.17), in which the algorithm always produces the token with the highest probability. This decoding algorithm usually results in repetitive and bland texts since it ignores other tokens with also high probability. Therefore, different alternative algorithms have been proposed in the literature to produce more human-like texts with rich content. For example, while decoding, the *beam search* algorithm keeps track of the N sequences with the highest probability and finally selects only top one. Holtzman et al. (2019) proposed the *Nucleus Sampling* decoding algorithm that randomly samples a token from the conditional probability distribution of the trained model (Equation 2.18).

$$\text{Greedy Search:} \qquad y_t = \text{argmax}_y \, P(y|y_{1:t-1}) \tag{2.17}$$

$$\text{Nucleus Sampling:} \qquad y_t \sim P(y|y_{1:t-1}) \tag{2.18}$$

Further improvements to the nucleus sampling algorithm, such as *top-k* and *top-p* sampling, have been proposed. In the *top-k* sampling, only the most likely *top-k* tokens are used (Fan et al., 2018), while in *top-p* sampling, the algorithm selects the most likely K tokens that cover a probability mass of p as candidates for sampling.

Constrained text generation During decoding, one can impose certain constraints on the generated texts that ensure certain criteria. Holtzman et al. (2018) modify the decoding objective of the underlying generation language model (*LM*) (Equation 2.19) to include a mixture of k classifiers (scoring functions) each ensures that the text follows a specific criterion:

$$p(y, x) = log(p_{model}(y|x)) + \sum_k \lambda_k s_k(x, y) \tag{2.19}$$

Dathathri et al. (2020) proposed an algorithm that allows constraining the generated text by providing a bag-of-words (BOW) or a classifier, ensuring that the text

adheres to certain attributes. During decoding, to produce a new token, the algorithm first performs a forward pass through the model to compute the new hidden state H_{t+1} and a distribution over the vocabulary o_{t+1} based on the old history H_t. Then an update to the hidden state δH_t is computed based on the gradients between the probability distribution o_{t+1} and the desired one inferred from the BOW or the classifier. Finally, a new modified distribution is computed and used to sample a new token. We use this algorithm in our work to generate claims constrained for a specific audience (Section 4.2).

2.2 Argumentation

According to Van Eemeren and Grootendorst (2004), argumentation can be defined as follows:

Argumentation is a verbal, social, and rational activity aimed at convincing a reasonable critic of the acceptability of a standpoint by putting forward a constellation of propositions justifying or refuting the proposition expressed in the standpoint.

Stede and Schneider (2018) decompose this definition into the following basic aspects to be consider when analyzing natural language arguments. First, it is a *verbal and social activity*, which means that we are dealing with a linguistic phenomenon of social nature since multiple human parties get involved and might be affected by this activity. Second, in argumentation, we have a main *standpoint* around which the controversy arises - sometimes called *the main thesis*. As part of this activity, each side puts forward a *reasonable* argument composed of *a constellation of propositions* to support *pro stance* or refute *con stance* the main standpoint. In the following, we will give a short overview of different argumentation frameworks that model the composition and the interaction between arguments, then discuss the role of the audience in argumentation as one of the social aspects of this activity.

2.2.1 Argumentation Models

To enable machines to process and represent human argumentation, a dedicated line of research proposed frameworks to either model the internal structure of an argument like Toulmin's model (Toulmin, 1958), or the interaction between different arguments like Dung's model (Dung, 1995). As illustrated in Figure 2.4, the former decomposes an argument into three main components: *claim*, *grounds*, and *warrant*. The claim is the main statement that the argument is making, while the grounds are the set of facts that supports the claim through implicit or explicit assumptions called the warrant. Additionally, the framework considers three other components of an argument: *backing*, which are extra information to support the

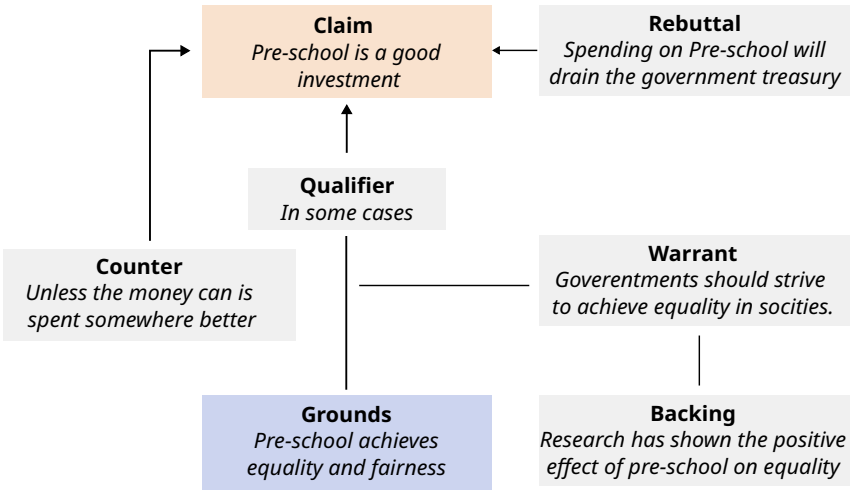

FIGURE 2.4: Example argument represented using Toulmin's argumentation model (Toulmin, 1958). In our work, we consider mainly the *Claim* and the *Grounds* components.

warrant, *qualifier* specifying certain situations in which the claim might not hold, and *rebuttal*, an acknowledgment of the existence of other counter-arguments.

Dung's framework models argumentation as a set of abstract arguments with attack relations between them. This framework represents argumentation as a graph on which computers can perform various tasks, for example, assessing the acceptance of an argument or inferring new relations. Several extensions to this framework are considered. For example, Cayrol and Lagasquie-Schiex (2005) extended the framework by considering also support relations between arguments. Skiba et al. (2020) considered a set of pieces of evidence to be required in order for an argument to appear in the argumentation framework. Gordon and Walton (2006) consider the internal structure of each argument in the argumentation graph as a statement with a set of supporting premises, and incorporate this internal structure in the overall assessment of the argument's strength. The value-based argumentation framework (Bench-Capon et al., 2002) is also an extension of Dung's framework that introduced a new notion to represent the audience's preferences as a set of human values. Accordingly, the acceptance of an argument is computed subject to a predefined set of values.

However, as modeled by these frameworks, the argumentation structure in daily life conversations is only sometimes apparent. In student essays (Stab and Gurevych, 2014a), one might find a clear formulation of the main thesis and supporting premises. However, the structure becomes unclear in other genres, such as online forums and news editorials. Since, in this thesis, we are dealing with primarily daily life argumentation, we follow a simple model of argumentation in which

we consider an argument to be consisting of a claim that we call a conclusion and a set of supporting and/or opposing premises.

2.2.2 Audience in Argumentation

As mentioned, argumentation is a social activity through which ideas are communicated and reasoned about between individuals or groups to reach an agreement. These different groups might vary in their beliefs, wishes, and interests. Achieving agreement among them requires finding common ground in the form of shared beliefs and using that as the means of communication.

In the argumentation field, Perelman (1971) and van Eemeren and Houtlosser (1999) advocated the important role of the audience in argumentation. Not only the argument's reasoning and structure are the factors of its success, but also the kind of audience it addresses. According to van Eemeren and Houtlosser (1999), one can use different rhetorical moves to resolve conflicts in opinions. These rhetorical moves can be categorized into three main aspects: (1) Relevant discussion points (*topic potential*), (2) ways of presenting the content (*presentational devices*), and (3) understanding the characteristics of an audience (*audience demand*). For example, in a debate on former US president Donald Trump, potential topics could have been immigration, health care plans, tax plans, etc. However, knowledge about the audience being middle-class workers would have suggested restricting the selection to Trump's tax plans. An appropriate usage of presentational devices may have then put a con argument as follows:

Example: "Donald Trump was a bad president. He did nothing but hurt the poor and middle class. His tax plan benefited only rich people who could afford it."

In the social science field, a body of research investigates the mechanisms of human judgment to understand the reasons behind disagreement. The moral foundation theory (Haidt and Joseph, 2004) offers a conceptual model of this disagreement. According to this theory, humans subconsciously adhere to five basic moral foundations when judging controversial issues: *fairness* (importance of justice, rights, and equality), *care* (being kind and avoiding harm), *loyalty* (self-sacrifice, solidarity, belongingness), *authority* (respect to traditions and hierarchy), and *purity* (sacredness of religion and human). Consequently, the disagreement between liberals and conservatives, for example, can be explained by the moral gap between the two parties. While liberals rely mainly on care and fairness (so-called *individualizing morals*) in their assessment of controversial issues, conservatives consider all moral foundations more evenly, somewhat skewed towards the *binding morals*, that is, loyalty, authority, and purity (Graham et al., 2009). Several studies provided evidence of the robustness of the moral foundation theory in understanding people's behaviors and decisions (Feinberg and Willer, 2015, Fulgoni et al., 2016). For example, Feinberg and Willer (2015) demonstrated how political arguments

tend to be created based on the moral views of its author, rather than considering the others' perspective. Due to this moral gap, these arguments are ineffective in resolving disagreements between disagreeing parties. When the authors re-framed these arguments to fit the moral perspective of the target audience, these arguments became more effective. Moreover, Schwartz (1994) proposed another perspective to model human beliefs by defining a taxonomy of universal aspects, called *Human Values*. In this taxonomy, one can identify values that conflict with each other such as *Achievement* and *Benevolence*, where the former relates to personal success and competition while the latter is about caring for the welfare of friends and family. These values provide a framework to examine the behavior of individuals and the dynamics of societies.

Audience in argumentation forms one of the main aspects we study in our thesis. We aim to integrate the audiences' beliefs into the process of argument generation to make these arguments more effective. We use the moral foundation theory as well as the audiences' stances on known issues to model these beliefs.

2.3 Computational Argumentation

As mentioned, our research builds on and contributes to the field of computational argumentation. In the following, we will present state-of-the-art methods for analyzing and synthesizing arguments in natural language texts relevant to our research, as well as applications in which these methods are used.

2.3.1 Argument Analysis

Mining Argumentative Structures The automatic analysis of argumentative structures in natural language texts is one of the main tasks in computational argumentation (Stede and Schneider, 2018). Researchers have investigated different genres, such as student essays, online forums, and news editorials. Typically, given an input text, argumentative spans of text are first identified (Ajjour et al., 2017). Then, each argumentative span (unit) is classified into a premise, claim, or main claim (conclusion), and relations between these units are identified as support or oppose stances (Stab and Gurevych, 2014b). Several approaches addressed this task either in monological texts such as student essays Stab and Gurevych (2014b) or in a dialogical setting such as online persuasive forums Chakrabarty et al. (2019b). Another body of research focused on retrieving argumentative texts relevant to a given topic from big corpora. Levy et al. (2018) constructed a sentence-level corpus of arguments and proposed a deep neural network model for the task. Stab et al. (2018) proposed an annotation scheme applicable to mining arguments from heterogeneous sources and constructed a corpus of 25k instances covering eight topics. They proposed a BiLSTM model to retrieve argumentative texts for a given topic. Bar-Haim et al. (2017) addressed the task of stance identification between

context-independent claims mined from such corpora. Their approach first identifies the target phrase of each claim and the corresponding sentiment. Second, a semantic relation between the two target phrases is identified and used to infer a final stance score. Some of our methods in this work also extract the target phrase of a given claim - we learn this from the data provided by Bar-Haim et al. (2017) (Section 5.1.1).

Moreover, approaches to identifying conclusions in a text have been studied in student essay. Falakmasir et al. (2014) highlights the importance of essay conclusions in various applications, whereas Jabbari et al. (2016) specifically addressed an essay's main claim (*thesis*). In this thesis, however, we focus on arguments in which conclusions are left implicit. As Habernal and Gurevych (2015) observe, real-world arguments often leave the conclusion implicit, particularly where it is clear in the context of a discussion. In genres such as news editorials, conclusions may be left out on purpose to persuade readers in a "hidden" manner (Al-Khatib et al., 2016). Therefore, the task becomes more challenging and might require inference.

In our work, we start from the assumption that arguments do not necessarily state their conclusion explicitly. Therefore, we develop methods to infer conclusions of arguments to then use these conclusions to generate better counters for input arguments (Section 5.2.2).

Assessing Argument Quality A significant body of research has studied argument quality assessment. Wachsmuth et al. (2017c) proposed a taxonomy of 15 quality dimensions that cover logical, rhetorical, and dialectical aspects of argument quality and later studied the automatic prediction of these quality dimensions (Wachsmuth and Werner, 2020). Both Stab and Gurevych (2017) and Gurcke et al. (2021) studied the prediction of quality scores that reflect whether given premises are sufficient to the conclusion. On the contrary, Gretz et al. (2020b) follows a holistic approach to the task in which an overall quality score is learned from the data. To this end, the authors build a corpus of pairs of arguments that are ranked based on their quality and present a neural method to learn to rank arguments based on their quality. On a more fine-grained level, given a set of premises and a conclusion, Jo et al. (2020) studied the task of premise attackability, which implicitly resembles the premise's acceptability dimension in Wachsmuth et al. (2017c). The authors learn this criterion from the sentences of posts that users attack in the Reddit forum *"Change My View (CMV)"*. These sentences represent premises supporting the claim encoded in a post's title. The authors experimented with different features that reflect weaknesses in the premises. Their best model for identifying attackable premises is a BERT-based classifier. Our approach also identifies the attackability of the argument's premises to counter it effectively, but we model this task as a ranking rather than a classification task.

Besides its reasonableness, the strength of an argument also depends on other rhetoric dimensions utilized in the argument, such as style, evoked emotions, etc. (Wachsmuth et al., 2018a, Wiegmann et al., 2022). Nevertheless, as mentioned earlier, the effectiveness of these strategies depends much on the targeted audience. What is effective for a middle-class crowd of workers might not be effective for a crowd of worshipers in a church. Therefore, researchers also considered audience-based features to model the persuasiveness of arguments. Among these, both Durmus and Cardie (2018) and El Baff et al. (2020) study how user factors such as religion and political background affect persuasiveness. Al Khatib et al. (2020) demonstrated that user-based features reflecting beliefs, characteristics, and personality could increase the predictability of argument persuasiveness. Moreover, Lukin et al. (2017) demonstrated that persuasiveness correlates with users' personality traits. Our approach to effective argument generation draws from this mount evidence on the importance of audience in argument generation. We follow El Baff et al. (2018) by considering an effective argument to be an argument that challenges its audience if they have a different stance or empowers them in case of a similar one. By generating arguments that challenge the two opposing parties, one could widen their minds to different perspectives, which brings them closer to agreement.

2.3.2 Argument Generation

Argument Summarization Summarization approaches aim to compress input text into a shorter version while conserving the most salient information in the original text. These approaches are categorized based on the inputs and outputs or by the utilized methods. In terms of output, summarization can be either *extractive* or *abstractive* (Gholamrezazadeh et al., 2009). In extractive summarization, the summary is an extract from the original text that carries the most important information. In contrast, abstractive summarization approaches synthesize new texts that highlight the core information of the input text. Regarding the input type, summarization can be applied into *single-document* or *multi-documents*.

Initially, summarization techniques relied on graph-based approaches. Here, the input text is split into a set of sentences that serve as nodes in a graph. These nodes are connected to each other by edges based on either lexical or semantic similarity sim. An unsupervised graph algorithm like PageRank (Page et al., 1999) is then employed to assign importance scores to each sentence, as shown in equation 2.20. The importance score of sentence s_i is computed based on all its similar sentences $M(s_i)$ and their similarity to s_i and importance score. The damping factor d is used for the convergence of the method.

$$P(s_i) = \frac{d}{N} + (1-d) \cdot \sum_{s_j \in M(s_i)} \frac{sim(s_i, s_j)}{\sum_{s_z \in M(s_j)} sim(s_z, s_j)} \cdot P(p_j) \tag{2.20}$$

Finally, the summary is produced by selecting the top N sentences (Erkan and Radev, 2004). However, with the increasing availability of data, neural models are now being employed to learn in a supervised manner and generate either extractive or abstractive summaries. A recent survey of popular neural approaches to summarization can be found in Hou et al. (2021). For instance, Liu (2019) fine-tuned the BERT model Devlin et al. (2018) on the CNN/Dailymail (Hermann et al., 2015) news summarization dataset and achieved state-of-the-art results. We reuse the work of Erkan and Radev (2004) and Liu (2019) as baselines for the task of modeling key points candidacy and compare them to our proposed approach (Section 3.1). One area of research in summarization is comparative summarization, which involves generating summaries that help in comparing the differences between document groups (Wang et al., 2013, Li et al., 2009). We adapt a recent approach for comparative summarization (Bista et al., 2020) to our use case in which we model the contrastiveness aspect of key points (Section **??**), where our objective is to obtain snippets that highlight the distinctions between arguments rather than groups of documents.

Compared to other document types, summarizing arguments is an understudied topic, potentially due to the absence of data. However, in the last years, a few approaches have been proposed that can also be categorized into single and multi-document summarization. When considering single arguments, approaches to mining an argument's main claim (conclusion) can be considered extractive summarization (Petasis and Karkaletsis, 2016, Daxenberger et al., 2017). Wang and Ling (2016) proposed a sequence-to-sequence model as an abstractive approach to summarize arguments from online debate portals. As for multi-document argument summarization, early approaches aimed to distill the main points in an online discussion. For example, Egan et al. (2016) grouped verb frames into clusters that serve a summarization pipeline. Misra et al. (2016) proposed a more focused approach by directly extracting argumentative sentences, summarized by similarity clustering. However, recently, Bar-Haim et al. (2020a) introduced the notion of *key points*, defined as concise and self-contained argumentative statements. They constructed a corpus of arguments mapped to manually-curated key points. Later, Bar-Haim et al. (2020b) proposed a quantitative argument summarization framework that automatically extracts these key points from a collection of arguments. Our approach to constructing the discussion space of a topic builds on this idea, in which we propose a method to better match arguments with key points.

Argument Component Synthesis As mentioned, often certain argument components such as premises or conclusions are not explicitly mentioned in the argument because they can be inferred from the context or because of rhetorical reasons (Habernal and Gurevych, 2015, Al-Khatib et al., 2016). This phenomenon motivated a line of research to work on reconstructing missing argument components. For example, Boltuzic and Šnajder (2016) study the task of enthymemes reconstruction. Similarly, Rajendran et al. (2016) aim to generate the premise connecting an aspect-related opinion to an overall opinion. Recently, Habernal et al. (2018) presented the task of identifying the correct missing warrant of an argument from two options and constructed a dataset to study this task. Chakrabarty et al. (2021) utilized commonsense knowledge to improve the performance of Seq2seq models on the task of generating missing premises. However, our approach to generating relevant counter-arguments relies on correctly inferring the argument's conclusion. Hence, we focus on studying the task of conclusion inference in scenarios where the conclusion is not explicitly mentioned in the argument.

Argumentative Text Generation Early research on argument generation aimed to create argumentative texts given a symbolic representation (Zukerman et al., 2000, Grasso et al., 2000, Carenini and Moore, 2006). These approaches had a similar architecture consisting of three main phases: text planning, sentence planning, and realization (Stede and Schneider, 2018). Nevertheless, they were applied on a limited scale. Nevertheless, recent advances in the NLP and machine learning fields have led to more research addressing a variety of argument generation tasks, such as argument generation for a given topic, countering an input claim or an input argument, generating arguments addressing specific aspects, etc. These approaches are either retrieval-based or generation-based. For example, Sato et al. (2015) proposed an approach to argument generation based on sentence retrieval, in which, given a topic, a set of paragraphs covering different aspects is generated. El Baff et al. (2019) used a language model approach to select and arrange a set of argument components into a full argument representing a certain argumentative strategy. To gain more control over the aspects addressed in the generated arguments, Schiller et al. (2020) proposed an approach that utilizes a pre-trained language model (Keskar et al., 2019) to generate arguments on a specific topic with a controlled stance and aspect. More recently, Al Khatib et al. (2021) used argumentation-related knowledge graphs constructed in their previous work (Al-Khatib et al., 2020) to control the output of GPT-2. However, to our knowledge, controlling the generated arguments to address a specific audience representation has never been studied so far. This thesis introduces the audience-based argument generation task and proposes approaches to frame the generated arguments based on a given audience representation.

To generate counter-arguments, Orbach et al. (2020) proposed a retrieval-based approach that retrieves relevant counters for a given argument from a collection of documents. Similarly, Wachsmuth et al. (2018b) utilized topic knowledge to retrieve the best counter for a given argument. In contrast, both Bilu et al. (2015) and Hidey and McKeown (2019) proposed generation-based approaches to counter-claim generation. The former developed a set of rules and classifiers to negate claims, while the latter used neural methods to learn from data. Moreover, Hua and Wang (2018, 2019) proposed an approach for generating long texts and applied it to the counter-argument generation task. Their approach relies on a retrieval component that acquires relevant key phrases for an input argument to be used to guide the generation of counter-arguments. While the size of the given argument collection limits retrieval-based approaches, the generation-based approaches either rely on the conclusion being given in the input or do not distinguish the different components in the input argumentative text. Our proposed approach is generation-based, where we study the conclusion's role in counter-argument generation.

2.3.3 Applications

The ultimate goal of the research in computational argumentation is to build tools that assist humans in daily life tasks such as making informed decisions concerning controversial topics, constructively participating in discussions, and resolving conflicts between disagreeing parties. Applications like argument search and debate technologies benefit directly from computational argument research. In the following, we will provide an overview of these two applications since our thesis take them as an application scenario.

Argument Search Engines Generally, search engines users my have various search goals that can be categorized into *navigational*, *informational*, and *resource* (Rose and Levinson, 2004). Several experiments demonstrated the importance of snippets for search engine usability to achieve these goals (Marcos et al., 2015). Therefore, search engines present search results along with short text *snippets* to help users in assessing the relevance of the underlying document (Croft et al., 2009). Usually, a snippet shows a representative excerpt of the web page's content, ideally including all query terms (Croft et al., 2009). This is a suitable compromise for the varying search goals that users may have (Rose and Levinson, 2004) when dealing with general-purpose search engines.

Recently, more attention has been given to argument retrieval from the web (Dumani and Schenkel, 2019, Potthast et al., 2019), and specialized search engines have been proposed that aim to give an efficient overview of the best arguments on a controversial issue queried (Stab et al., 2018, Wachsmuth et al., 2017b). As illustrated in Figure 2.5, given *school subsidy* as a queried controversial topic, a set of arguments is retrieved and split into two columns representing the pro and

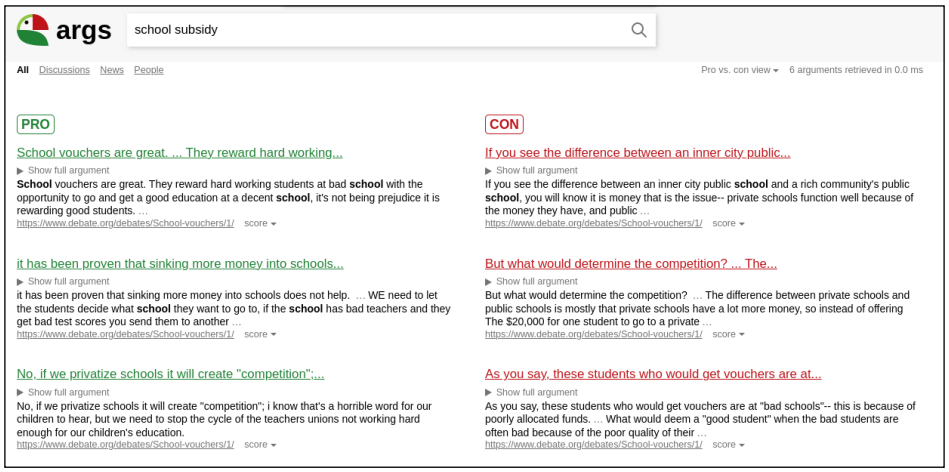

FIGURE 2.5: A screenshot of the *args.me* argument search engine. Given a topic *school subsidy*, the engine retrieves a set of pro and con arguments relevant to the topic.

con sides of the controversy. Similar to general-purpose search engines, snippets here are also essential to give an efficient overview of the gist of each presented argument. Nevertheless, work in the field still needs to address this task. As part of our approach to identify the discussion space of a given topic, we study the task of extracting argument snippets in the argument search scenario. These snippets can then be also aggregated and presented as key points of the discussion space.

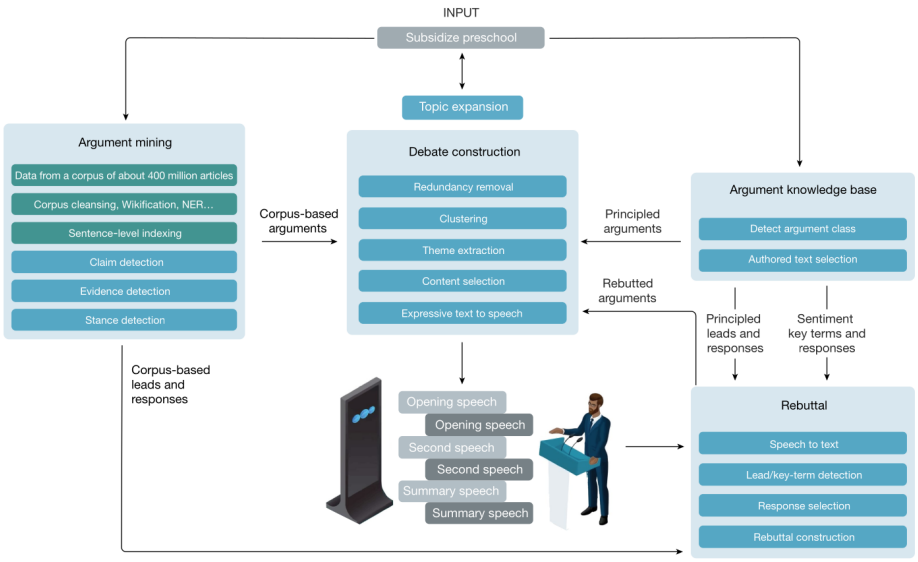

FIGURE 2.6: Project Debater's components and their interactions (Slonim et al., 2021). Given a topic, several components perform various computational argumentation tasks such as claim and evidence detection, stance classification, and rebuttal generation.

Debate Technologies One can consider debate technologies as decision assistant systems that help users make informed decisions concerning a stance towards a particular claim or topic, for example, whether to support *school subsidy*. Debate technologies can assist humans by empowering their position via supporting arguments or challenging it through counter-arguments. This assessment is not straightforward since it requires a set of capabilities such as natural language understanding and synthesis, as well as evidence mining and retrieval. Despite these challenges, recently, Slonim et al. (2021) published a paper describing *Project Debater*, a system that can autonomously engage in debates on controversial topics with humans. This system is a result of seven years of research and consists of a set of components that perform various computational argumentation tasks. As shown in Figure 2.6, given an input topic and a stance, relying on a predefined corpus of arguments, the system can synthesize new arguments on the topic with the given stance and generate a counter for potential arguments that oppose this stance.

In particular, Slonim et al. (2021) address the task via an approach composed of four main components: argument mining, argument knowledge base, debate construction, and rebuttal. In the argument mining component, a huge collection of newspaper articles are processed into sentences. This collection is later used to perform claim and evidence detection for a given input topic (Bar-Haim et al., 2017). The argument knowledge base represents a compilation of principled arguments that can be used in a wide range of debates (Bilu et al., 2019). Given a topic, this component can find a matching principled argument that fits the topic. The debate construction component is then responsible for retrieving relevant claims and evidence from the argument mining component, finding the matching principled arguments, clustering them into themes, and synthesizing the final argument on a given topic. Finally, the rebuttal component is responsible for identifying the main claims in the opponent's argument and retrieving relevant counter-arguments from the argument mining and argument knowledge base components (Orbach et al., 2019).

In our thesis, we take a similar debate scenario as an example application to demonstrate the importance of our research questions. We identify different research areas to study to improve debate technology's engagement. Particularly, we argue that synthesized arguments need to consider the audience, a factor that is not considered in Project Debater. Second, instead of being bound to a predefined set of arguments to choose from, we study the task of generating counter-arguments from scratch.

Chapter 3

Modeling the Discussion Space

As discussed in Chapter 1, the first task a debate technology needs to address is identifying relevant key points to the input topic. These key points form a discussion space that can be used to guide the content of synthesized arguments. We decompose this task into (1) retrieving a set of relevant arguments and then (2) summarizing them by extracting salient key points. Since the first step is well addressed in research (Wachsmuth et al., 2017b, Stab et al., 2018), we focus on key points identification from a relevant set of arguments. As mentioned in the introduction, our main research question in this chapter is then: *How to effectively extract relevant key points for a topic that form its discussion space?*

Following Bar-Haim et al. (2020b), we define the discussion space of a topic as a set of concise statements representing high-level arguments that people bring up when discussing the topic. These statements are called key points hereafter. We build on the assumption that every sentence in the argument collection can be considered a candidate key point, and the task is then to find a set of sentences that best form the final set of key points. In particular, as illustrated in Figure 3.1, our approach first models the candidacy of sentences in the argument collection to be key points. Then it aggregates them into a final set of key points.

In particular, our approach's key point modeling component derives the candidacy score based on the argumentativeness and centrality of the sentence in the argument collection (context). The second component then aggregates the candidate sentences into the final set by first ranking them according to their scores and then selecting a diverse set of top k sentences that represent the collection and cover most perspectives in it. We call this approach the *Representativeness Approach*. Moreover, in a follow-up study, based on Alshomary et al. (2022b), we propose another method to diversify key points by adding the contrastiveness criteria into the notion of key point candidacy to ensure diversity. Here, in the first component of our approach, we compute scores for each sentence that consider the representativeness of the sentence of its own argument, its argumentativeness, and its dissimilarity to other arguments in the collection.

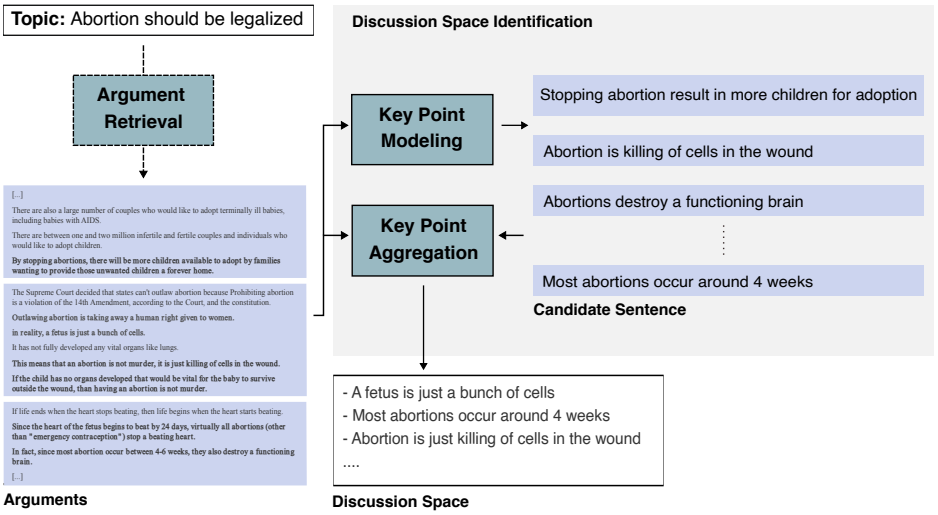

FIGURE 3.1: Our approach to the task of identifying the discussion space: Given a topic, we assume the existence of a component that retrieves relevant arguments to the given topic, and we focus on how to construct the discussion space from this collection. The first step is then to model the key point candidacy on the sentence level. Second, the scored sentences are aggregated into the final set of key points that represent the discussion space.

We evaluate our approach in two stages. First, we assess the performance of the candidacy modeling approach on a proxy task, namely *argument snippet generation*. In argument search, a snippet is a short excerpt that helps users assess the argument's relevance to their argument search goal. We consider sentences that form a good snippet of an argument to be valuable key points. We found out that our approach generates more useful snippets than corresponding baselines. Hence, these snippets can also be used as candidate key points. We additionally show in our experiments that adding the contrastiveness criteria into the key point candidacy equation can lead to a trade-off between the representativeness and contrastiveness of snippets, which one can configure according to the use case at hand.

In the second stage, since our task closely resembles the key point analysis shared task proposed by Friedman et al. (2021b), we evaluate our overall approach by participating in their shared task proposed at the 8th Workshop on Argument Mining (ArgMining 2021) at EMNLP 2021. The manual evaluation results reveal that our approach ranked best among other submitted approaches to this task. In follow-up experiments, we also compare our approach to Large Language Models (LLMs), namely ChatGPT, to generate key points. Despite the simplicity of our approach, it achieves comparable results while being interpretable. For example, we can explain why our approach selected a specific sentence as a key point by presenting its centrality in the modeled graph and argumentative scores. Our

Topic abortion
Argument The Supreme Court decided that states can't outlaw *abortion* because Prohibiting *abortion* is a violation of the 14th Amendment, according to the Court, and the constitution. Outlawing *abortion* is taking away a human right given to women. In reality, a fetus is just a bunch of cells. It has not fully developed any vital organs like lungs. This means that an *abortion* is not murder, it is just killing of cells in the wound. If the child has no organs developed that would be vital for the baby to survive outside the wound, than having an *abortion* is not murder.

FIGURE 3.2: Example argument for the topic "abortion" taken from Alshomary et al. (2020a). The underlined sentences are example of good candidate key points to be extracted from this argument

main findings are the following: (1) Argumentativeness and representativeness are important criteria to model when extracting key points from a collection of arguments, (2) Modeling contrastiveness of sentences can help extract a diverse set of key points, and (3) Modeling argumentativeness and representativeness of sentences can also be used to construct useful argument snippets for argument search.

In the following, Section 3.1 will first present our implementation of the key point candidacy modeling component. Second, in Section 3.2, we will describe the key point aggregation component and how to achieve diversity in among key points. We will discuss the aggregation algorithm and introduce our method to ensure diversity by encoding contrastiveness as a criterion of key point candidacy. Finally, we will present a series of experiments to evaluate our approaches in Section 3.3, including the argument snippet generation study (Subsection 3.3.1) and the overall evaluation of our approach to the key point analysis task (Subsection 3.3.2).

3.1 Modeling Key Point Candidacy

Our main task is to distill a set of key points from a given argument set relevant to the topic. We define this task's input as a set of $k \geq 2$ arguments $\mathbf{A} = \{A_1, \ldots, A_k\}$ relevant to the same discussion topic. We represent each $A \in \mathbf{A}$ simply as a set of sentences, $A := \{s_1, \ldots, s_n\}$, where $n \geq 2$ usually differs across arguments. The output is then a subset of all sentences in the collection $S \subseteq \cup_{A \in \mathbf{A}} A$ that forms the final set of key points. Accordingly, our approach has two components, as illustrated in Figure 3.1. The first component assigns key point candidacy scores for each sentence, and the second distills a final set of sentences to represent the discussion space, given their candidacy scores. This section will discuss the implementation of our approach's first component.

We define key points as high-level arguments that are prominent when the corresponding topic is discussed. Therefore, we hypothesize that two criteria influence the candidacy of a sentence in the argument collection to be a key point: central-

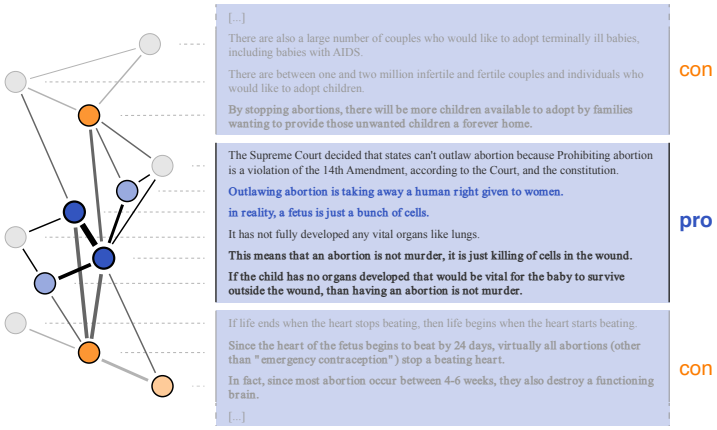

FIGURE 3.3: Illustration of our sentence scoring graph method for a pro argument on a web page discussing abortion (Alshomary et al., 2020a). Nodes with black borders represent argumentative sentences (bold text). Edge thickness reflects similarity and bias towards argumentativeness (very thin edges are omitted for a lean visualization). The more saturated a node, the higher the score.

ity and argumentativeness. For example, as highlighted in Figure 3.2, candidate sentences from the given argument on abortion can be the underlined ones highlighted in blue. To operationalize these criteria, as illustrated in Figure 3.3, we propose a graph-based approach following Erkan and Radev (2004) utilizing sentence embeddings to capture the semantic similarity between sentences and biasing our model towards sentences with argumentative language.

Accordingly, given the set of arguments **A** relevant to an input topic, we split them into their sentences, and for each sentence s, we compute an embedding $e(s)$ and an argumentative score $arg(s)$. The resulting graph covers all sentences (represented by their embedding) as nodes, and the similarity $sim(e(s_i), e(s_j))$ of the embeddings represents the edge weight between s_i and s_j. Finally, we apply the PageRank (Page et al., 1999) algorithm to generate an importance score $P(s_i)$ for each sentence s_i:

$$P(s_i) \;=\; (1-d)\cdot \sum_{s_j \neq s_i} \frac{sim(e(s_i), e(s_j))}{\sum_{s_k \neq s_j} sim(e(s_j), e(s_k))} P(s_j) \;+\; d \cdot \frac{arg(s_i)}{\sum_{s_k} arg(s_k)}$$

As the equation captures, $P(s_i)$ is a sum of two parts, weighted by a damping factor d. The first part reflects the centrality of s_i, computed concerning the importance $P(s_j)$ of other sentences s_j and their similarity to s_i. The second part represents a normalized bias towards the argumentativeness nature of s_i, which can be computed by utilizing argument mining techniques. Figure 3.3 illustrates the influence of similarity and argumentativeness. As in the original PageRank, we start with equal scores for all sentences and iteratively update them until near-convergence. These final scores represent then the key point candidacy of all sen-

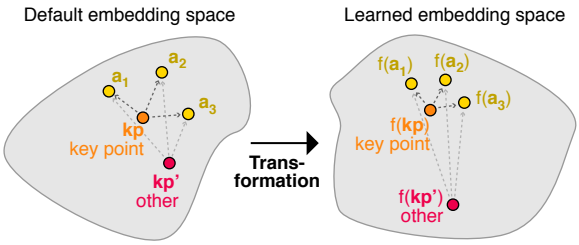

FIGURE 3.4: Illustration taken from Alshomary et al. (2021b). We learn to transform an embedding space into a new space in which matching pairs of key point and argument (e.g., kp and a_1) are closer to each other, and the distance between non-matching pairs (e.g., kp' and a_1) is larger. For simplicity, kp and kp' each represent a concatenation of a key point and its topic..

tences in the collection based on which the aggregation of the final set can be performed. In our experiments, we call this approach *Arg-PageRank*.

Learned Representation

Recall that we modeled the centrality of sentences by their similarity to others in the embedding space, which we can compute via various embedding methods e that capture semantics of input sequences (Reimers and Gurevych, 2019). Nevertheless, one can also learn a more domain-specific embedding that captures the argument key point relation. Given a dataset of pairs of arguments and key points, we seek to transform the generic embedding space into a learned one where matching pairs are closer, and the non-matching ones are more distant (Figure 3.4). We utilize a siamese neural network (Bromley et al., 1994) with a contrastive loss to learn this mapping function.

Specifically, in the training phase, the input to our model is a key point, an argument, and a label (matching or not). First, we use a pre-trained language model to encode the tokens of the argument and the key point. Then, we pass their embeddings through a siamese neural network, which is a mean-pooling layer that aggregates the token embeddings of each input, resulting in two sentence-level embeddings. We compute the contrastive loss using these embeddings as follows:

$$\mathcal{L} \;=\; -y \cdot \log(\hat{y}) \;+\; (1-y) \cdot \log(1-\hat{y})$$

where $\hat{y}$ is the cosine similarity of the embeddings, and y reflects whether a pair matches (1) or not (0). We can then use the trained model to compute matching scores between sentences reflecting our graph's edge weight. This step is optional and can be executed if we have a data set to learn this relation.

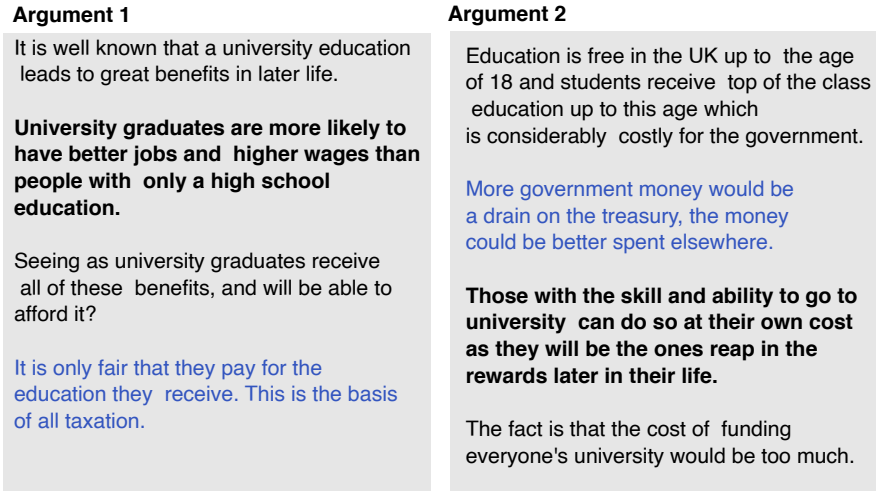

FIGURE 3.5: Example of two arguments on the topic "tuition fees", taken from Alshomary et al. (2022b). In each argument, the bold sentences are representative of their own argument but redundant when jointly summarizing the two arguments. The blue highlighted sentences are better choices since they highlight the unique aspect of their own argument (contrastive) while still being representative.

3.2 Key Point Aggregation

The input to this component is a set of candidate sentences, each with its own candidacy score. The goal is then to find a diverse set of k sentences that best represent the collection but also cover most of the perspectives mentioned in it. In this section, we first shortly explain our rule-based algorithm to aggregate these key points. We then propose a method to ensure diversity by encoding contrastiveness as an extra criterion into the key point candidacy modeling.

3.2.1 Rule-based Algorithm

Our key point aggregation algorithm ranks the scored sentences in descending order based on their candidacy scores. Then, it iterates through the ranked list, adding sentences to the final set based on semantic similarity. In particular, given a sentence, the collection of already added key points, and a threshold τ, we first compute the similarity score between the sentence and every key point in the collection. If the maximum computed similarity is below the given threshold τ, we add the sentence to the collection. Otherwise, we do not add the sentence. We repeat the process for each sentence until we have a k sentence in our collection.

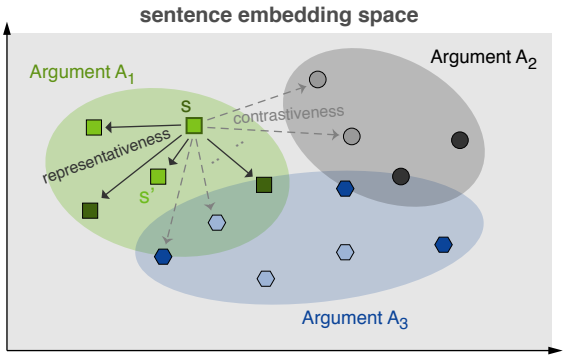

FIGURE 3.6: The idea of both realizations of our Contrastive Approach (Alshomary et al., 2022b), illustrated for three arguments in a sentence embedding space. A sentence s is considered important if its joint representativeness and contrastiveness are higher than for other sentences s' of the same argument. Argumentativeness (brighter symbols) is considered as well.

3.2.2 Modeling Contrastiveness

So far, our proposed approach modeled two criteria of sentences: *representativeness* and *argumentativeness*. However, we might extract redundant key points by focusing only on representativeness. Figure 3.5 demonstrates an example of this scenario. Given two arguments on topic *tuition fees*, by extracting only representative sentences, we might end up with the bold highlighted sentences as key points, which are redundant. Therefore, to ensure the diversity of extracted key points, we propose additionally considering the *contrastiveness* when computing the key point candidacy. Specifically, given the input defined as a set of arguments $\mathbf{A} = \{A_1, \ldots, A_k\}$, we aim to increase the importance of sentences that are representative of their argument, argumentative, and contrastive towards all arguments in $\mathbf{A} \setminus \{A\}$. In our example (Figure 3.5), the blue highlighted sentences become better candidate key points.

The following will describe our two alternative implementations of this new criteria. The first, *Contra-PageRank*, extends our previous approach by modeling the dissimilarity of each sentence $s_i \in A$ to all sentences from $\mathbf{A} \setminus \{A\}$. The second, *Comp-Summarizer*, adapts the work of (Bista et al., 2020) to select a sentence s_i that can be inferred that it comes from $\mathbf{A} \setminus \{A\}$ but not from A. As illustrated in Figure 3.6, both thus follow the idea to value sentences that are both representative of their own argument A and contrastive to other arguments $\mathbf{A} \setminus \{A\}$. Notice that in the Representativeness Approach, the argument collection is used to compute how representative a sentence is of the whole collection. In contrast, here, the representativeness of a sentence is considered only concerning its own argument. Our Contrastiveness approach then uses the argument collection to compute only contrastiveness scores.

Contra-PageRank In Subsection 3.1, we discussed our graph-based approach that utilizes PageRank (Page et al., 1999) to score sentences in terms of their centrality in context and argumentativeness. In order to extend this to model contrastiveness, we modify the underlying scoring function $P(s_i)$ in two ways: First, we compute the centrality of a sentence $s_i \in A$ based only on the sentences in its covering argument A only rather than all sentences from the whole context of $\mathbf{A}$—to avoid conflicts with our second adaptation. Second, we extend the bias term that represents the initial sentence probability to account not only for argumentativeness (arg) but also for contrastiveness. The contrastiveness here is quantified as a discount on the similarity (sim) of s_i to other arguments in the context. As a result, we reformulate the PageRank score of $s_i \in A$ as follows:

$$P(s_i) := \quad d_1 \cdot \sum_{s_j \in A, i \neq j} \frac{sim(e(s_i), e(s_j))}{\sum_{s_k \in A, j \neq k} sim(e(s_j), e(s_k))} \cdot P(s_j) \tag{3.1}$$

$$+ \quad d_2 \cdot \frac{arg(s_i)}{\sum_{s_j \in A} arg(s_j)} \quad - \quad d_3 \cdot \frac{sim(e(s_i), \mathbf{A} \setminus \{A\})}{\sum_{s_j \in A} sim(e(s_j), \mathbf{A} \setminus \{A\})} \tag{3.2}$$

Here, the argumentativeness score $arg(s_i)$ of each $s_i \in A$ and the similarity score $sim(s_i, \mathbf{A} \setminus \{A\})$ are computed directly to form the initial bias score of each sentence. Following our previous approach, a graph is then constructed for each argument A by modeling each sentence $s \in A$ as a node and creating an undirected edge $\{s, s'\}$ for each pair $s, s' \in A$, $s \neq s'$, weighted with $sim(e(s_i), e(s_j))$. Finally, PageRank is applied to generate a score $P(s)$ for each s.

Comp-Summarizer Given the resemblance of our task to comparative summarization, we model the task in line with the mentioned approach of Bista et al. (2020): For an argument A, the goal is to find sentences $S \subseteq A$ subject to (a) S being representative of A, and (b) S being contrastive to $\mathbf{A} \setminus \{A\}$. This goal is conceptualized via a condition for each objective: (a) No classifier y can be trained that distinguishes sentences in S from those in $A \setminus S$, reflecting representativeness. (b) A classifier y' can be trained that can differentiate sentences in S from those in other arguments from the context $\mathbf{A} \setminus \{A\}$, reflecting contrastiveness.

Regarding the classifiers, condition (a) aims to minimize the accuracy of y, whereas (b) aims to maximize the accuracy of y'. Since finding such classifiers is an intractable problem in general, Bista et al. (2019) used maximum mean discrepancy (MMD) (Gretton et al., 2012) as an estimation of the classifiers' effectiveness. The MMD algorithm is a method of computing distances between two probability distributions. Given a set of arguments $\mathbf{A}$, the goal is then to find sentences $S \subseteq A$ of all arguments $A \in \mathbf{A}$ that maximize the following term:

$$\sum_{A \in \mathbf{A}} \big(- MDD^2(S, \{A\}) + \lambda \cdot MMD^2(S, \mathbf{A} \setminus \{A\}) \big) \tag{3.3}$$

Here, λ is a parameter to control the influence of contrastiveness (second addend the term above). This formulation models representativeness based on sentence similarity (first addend). It can be solved in an unsupervised way by greedily selecting sentences that satisfy the objective.

However, other features may signal sentence importance that are not reflected by similarity (e.g., argumentativeness in our case). For this, Bista et al. (2020) introduced learnable functions that map sentence features into an importance score and integrate them into the objective function of a supervised MMD variant. Given a training set $\mathcal{T}$ with tuples of argument A, a set of sentences $\bar{S} \subseteq A$ that is a good representation of the argument, and context $\mathbf{A}$, the goal is to minimize (note the switched signs) the following adjusted term:

$$\frac{1}{|\mathcal{T}|} \sum_{(A, \bar{S}, \mathbf{A}) \in \mathcal{T}} \big(MDD^2(\bar{S}, A, \theta) - \lambda \cdot MMD^2(\bar{S}, \mathbf{A} \setminus \{A\}, \theta) \big) \tag{3.4}$$

Here, $\theta \in \mathbb{R}^m$ denotes a vector of learned feature weights. The adjusted variant requires the definition of sentence features that reflect its likelihood of appearing in $\bar{S}$. Hence, we consider the following $m = 6$ features in our implementation:

1. *Position.* Position of the sentence in the argument
2. *Word count.* Number of words in the sentence
3. *Noun count.* Number of nouns in the sentence
4. *TF-ISF.* TF-IDF on the sentence level
5. *LexRank.* Scores obtained from LexRank Erkan and Radev (2004)
6. *Argumentativeness.* Count of words from a claim lexicon, similar to our previous approach in Subsection 3.1

3.3 Evaluation

This section will present a series of experiments to evaluate our approach. In the first set of experiments, we consider sentences that form a useful snippet of an argument in argument search to be also good candidate key points. Therefore, we test our candidacy modeling approach on the argument snippet generation task and then evaluate the effect of adding contrastiveness as an extra criterion. We first introduce and motivate the argument snippet generation task and then present our

approaches' evaluation results. Second, we will provide details on the overall performance of our approach when tested on the task of key point analysis proposed by Friedman et al. (2021a), which resembles the idea of identifying discussion space. However, due to the chronological development of our research, we only experimented with our rule-based algorithm to aggregate the diverse set of key points, and we still need to test the effectiveness of modeling contrastiveness. Nevertheless, results from the argument snippet generation task demonstrate an added value by modeling contrastiveness. Future research should consider implementing this criterion to ensure the diversity of key points.

3.3.1 Argument Snippet Generation

An *argument snippet* can be a good candidate to represent the potential key points emerging in the discussion space of the topic. So, we consider evaluating our key point candidacy approaches on this task. In argument search scenario, general-purpose snippets that rely on extracting sentences overlapping with the submitted query are insufficient. As shown in Figure 3.2, a general-purpose snippet is likely based on the argument's beginning, which contains the query term (*Abortion*) multiple times. However, the main point the argument is about comes afterward, shown underlined in the figure. Hence, in Alshomary et al. (2020a), we argued that specific snippets are needed for arguments. To our knowledge, such argument snippets are not yet studied in existing research. Therefore, in the following, we present our practical definition of the argument snippet generation task.

Task The assumption behind argument search is that users who aim to form a stance on a controversial issue need an efficient overview of the most relevant arguments (Stab et al., 2018, Wachsmuth et al., 2017b). Ideally, we have a collection of arguments, each is composed of a (main) claim that has a stance on the issue or its aspects, along with a (main) reason supporting the claim (Stede and Schneider, 2018). Nevertheless, argumentative texts may phrase multiple claims and reasons, spread them over multiple sentences, add non-argumentative background information, all combined in various ways. To get the gist of an argument, a user needs to identify the actual inference from reason to claim, irrespective of the phrasing. Hence, we argue that an argument snippet should support this process. Accordingly, our working definition is that a good argument snippet represents the main claim and the main reason supporting it in a short summary. Since we expect that claim and reason can be expressed in a single sentence each and since two sentences roughly match what fits into the typical length of a search engine snippet, we restrict our view to two-sentence snippets, and we define the argument snippet generation task accordingly as:

> *Given a set of natural language arguments, generate for each argument a two-sentence snippet that best represents the argument's main claim and reason.*

In general, given a set of $k \geq 2$ arguments $\mathbf{A} = \{A_1, \ldots, A_k\}$ on the same topic, each is represented as a set of sentences, $A := \{s_1, \ldots, s_n\}$. The output is one subset $S \subseteq A$ for each A, consisting of all sentences of the snippet.

In the following, we will present the experiments that evaluate our approach to model key point candidacy in extracting representative snippets from arguments. We will then introduce the experiment setup to evaluate the contrastiveness extension of our approach in extracting snippets that are representative of the argument but also highlight its unique aspects compared to other arguments. The representativeness and contrastiveness criteria can be utilized for argument snippet generation differently. While the Representative Approach generates snippets for single arguments focusing on extracting their main points, extending it with the contrastiveness criteria leads to shifting the snippet toward unique aspects mentioned in the argument compared to others. Therefore, we evaluate each independently against its corresponding baselines in our experiments.

Modeling Representativeness

Data To evaluate our Representative approach, in Alshomary et al. (2020a), we constructed a first benchmark dataset with ground-truth snippets for a sample of arguments: To this end, we retrieved arguments for the 10 queries most often submitted to *args.me* (Ajjour et al., 2019b). All arguments come from debate-like web pages. Each has a stance towards an issue-like conclusion and a debate identifier. For each query, we took the top five pro and top five con arguments and filtered out all trivial cases, i.e., those with maximum two sentences. The length of the remaining 73 arguments ranges from 3 to 84 sentences with a median of 9. We asked two human experts to select the two sentences from each argument that, in their given ordering, define the most representative snippet according to our working definition. In 77% of the cases, the experts agreed on at least one sentence, with a Cohen's κ agreement of 0.50. Disagreement cases were resolved in onsite discussion between them, which worked out well in all cases. We call this dataset *expert-representative-snippets*. We randomly split the final dataset into 23 arguments for development and 50 for testing.

Implementation Details Given a collection of arguments, we first split them into their sentences using the NLTK library[1]. Sentences are then embedded using the universal sentence embedding model (Cer et al., 2018), and cosine similarity is computed between them. To compute the *argumentative score*, we use a discourse

[1] `https://www.nltk.org/`

#	Approach	Accuracy
b1	Random sentence selection	27%
b2	LexRank	27%
b2*	LexRank + weighted similarity	36%
b3	BertSum	33%
b3*	BertSum + optimized towards snippet extraction	40%
a	Arg-PageRank (centrality)	43%
a*	Arg-PageRank (centrality + argumentativeness)	**44%**

TABLE 3.1: Automatic evaluation: Accuracy of all evaluated snippet generation baselines and our approach variations in matching the ground-truth snippets of the given test set.

lexicon, constructed from discourse markers and claim words (Levy et al., 2017). We evaluate two variants: *(a) Arg-PageRank (centrality)* considers only the centrality of a sentence in its debate context ($d = 0$), and *(a*) Arg-PageRank (centrality + argumentativeness)* computes the argumentative score of a sentence as the frequency of discourse markers in it ($d = 0.15$, default value). In a follow up experiment (Section 3.3.1), we evaluate other argumentative scoring methods.

We compare our snippet generation approach to *(b_1) random sentence selection*, in order to assess what gains we achieve. In addition, we consider two extractive summarization baselines: On one hand, we use the Python implementation of the unsupervised graph-based method *(b_2) LexRank* (Erkan and Radev, 2004). As our approach, we apply it to all sentences of all arguments from the same debate. We create an unweighted edge when the similarity of two sentences exceeds 0.1. For further optimization, *(b_2^*) LexRank + weighted similarity* uses the similarities themselves as the weights of all respective edges. On the other hand, we train the transformer-based model *(b_3) BertSum* (Liu, 2019) in a supervised way to extract those two sentences that most resemble the argument's conclusion, since conclusions may retain representative information. We trained *BertSum* on the dataset from Wang and Ling (2016), which contains pairs of premises and conclusion from online debates, considering the conclusion as the summary of its premises. Since these conclusions are not always excerpts from the premises, we modified b_3 to *(b_3^*) BertSum + optimized towards snippet extraction*, such that the summary is given by the two sentences from premises that most overlap with the conclusion in content tokens, and trained it accordingly.

Automatic Evaluation We computed the accuracy of each approach in selecting the ground-truth snippets' sentences, averaged over all arguments in the dataset: For each argument, the accuracy is either 0 (no selected sentence correct), 0.5 (one correct), or 1 (both correct). Table 3.1 shows the results. While *LexRank* does not improve over *random sentence selection* (both 27%), the modification b_2^* improves it to 36%, and the optimized *BertSum* (b_3^*) even achieves 40%. Our approach *Arg-*

		Readability		Representativeness	
#	**Approach**	**Mean Rank**	**% Rank 1**	**Mean Rank**	**% Rank 1**
b2*	LexRank + w.s.	2.57	28%	2.47	28%
b3*	BertSum + o.t.s.e.	**2.15**	**46%**	2.43	26%
a*	Arg-PageRank (c.+a.)	2.50	26%	**1.95**	**44%**
	Expert snippets	1.71	66%	1.66	60%

TABLE 3.2: Manual evaluation: Mean readability and representativeness rank (lower is better) and proportion of best ranks (higher is better, multiple best ranks possible) of the test set snippets of selected approaches and of the expert snippets.

.

PageRank (centrality) already achieves higher accuracy than all baselines (43%). Encoding argumentativeness of a sentence (*a**) further increases accuracy, only slightly though (44%). We expect that the gain would be larger on datasets with fewer argumentative sentences, and with refined argument mining techniques. As mentioned, in a latter experiment we explore different methods of generating argumentative scores to gain more insights into this. Here, we conclude that modeling the centrality and argumentativeness of an argument's sentences turns out most successful in mimicking the ground-truth snippets.

Manual Evaluation To assess the quality of the generated snippets, we conducted two annotation studies, each with an independent set of three university students that have background on search engines and argumentation. Following literature, we mainly consider a snippet's *representativeness* (Liang et al., 2006) in terms of capturing the core information of the corresponding argument. In addition, we include *readability* (Kanungo and Orr, 2009) as another quality dimension, here meaning how coherent the two sentences of a snippet are on their own.

The first study compared our approach to the two best baselines and to the expert snippets. After explaining the task and quality dimensions, we showed the four competing snippets in random order for all 50 test arguments, and asked the annotators to rank the snippets according to both dimensions. To avoid bias, no training was done before. The mean pairwise inter-annotator agreement in terms of the rank-correlation measure Spearman's ρ was 0.52 for representativeness and 0.36 for readability, indicating general agreement but notable subjectivity in the given task. To give each annotator equal importance, we thus computed the mean ranks over all annotators. As Table 3.2 presents, *Arg-PageRank* clearly outperformed the others in terms of representativeness, being best in 44% of the cases and achieving a mean rank of 1.95. However, the readability of its snippets was judged worse than for *BertSum*. A reason may be that our approach tends to favor long sentences by concept. Besides, snippets generated by the experts proved best, underlining that our working definition is reasonable.

	Readability		Representativeness	
Approach	**Mean Rank**	**% Rank 1**	**Mean Rank**	**% Rank 1**
Lucene (query-dependent)	2.06	28%	2.17	22%
Args.me	**1.52**	48%	1.77	50%
Arg-PageRank (c.+a.)	1.60	**58%**	**1.69**	**60%**

TABLE 3.3: Second manual evaluation: Same as Table 2, but for snippets generated by the query-dependent Lucene algorithm, the search engine args.me, and our approach.

In the second study, we assessed whether our approach improves over approaches from practice. For this, we compared to the built-in snippet generation of *Lucene*, which is *query-dependent*: it selects text spans overlapping with query token. In addition, we evaluated the current snippets of *args.me*, which just show the beginning of arguments. All snippets were truncated after 225 characters, to mimic a real user-interface situation. We used the same setting as in the first user study but with different students to avoid bias. Spearman's ρ was 0.33 for representativeness and 0.36 for readability. Table 3.3 shows that *Arg-PageRank* again performs best in terms of representativeness, and is on par with the readability of *args.me*. The fact that our approach produces the most representative snippets in 60% of the cases provides empirical evidence for the need to address argument snippet generation as a special task, and highlights the limited of (at least standard) query-dependent snippet generation in argument search scenarios.

Modeling Contrastiveness

Based on our previous work (Alshomary et al., 2022b), we now provide empirical insights on the trade-off between representativeness and contrastiveness in snippet generation and how to adjust it via hyper-parameters tuning. In particular, we first explain the data collection and preprocessing we performed. We then provide details on implementing our approaches and evaluation measures used in our automatic assessment. We finally present the manual evaluation carried out to assess the generated snippets by our approaches compared to other baselines.

Data Since snippets generated by experts in Subsection 3.3.1 are not optimized toward being contrastive, we can't use them for our purpose. Therefore, we work on constructing a new dataset of arguments grouped into contexts (arguments relevant to the same topic), and use intrinsic evaluation measure for assessment. In particular, since the task here is motivated by the idea of argument search engines, we use the *args.me* corpus of Ajjour et al. (2019b) as the source. We considered all arguments in the corpus belonging to the same debate as a context, resulting in 5457 contexts with an average of 5.2 arguments per context, we call it *argsme* dataset. Such contexts suit the training of *Comp-Summarizer* since we can use

d_1	d_2	d_3	Contrastiveness	Argumentativeness	Representativeness
1.0	0.0	0.0	0.045	**0.647**	**0.800**
0.5	0.7	0.2	0.050	0.630	0.675
0.8	0.9	0.7	**0.060**	0.622	0.594

TABLE 3.4: Automatic evaluation scores of *Contra-PageRank* for three selected combinations of hyperparameter values. The best value in each column is marked in bold.

argument conclusions to derive generic snippets. Second, we mimicked how arguments are grouped into contexts in search by querying the *args.me* API[2] once using Wikipedia's list of controversial issues,[3] and once using queries from the *args.me* query log. We call the former dataset *controversial-contexts* containing 600 context with an average of 7.5 arguments per context, while the latter is called *query-log* containing 476 contexts with an average of 7.0 arguments. Since query-log is best in representing the realistic search scenario, we use it below for the final evaluation. We preprocess all input arguments in a number of cleansing steps. Namely, we remove debate artifacts that are mostly utterances of social interaction between debaters (Dorsch and Wachsmuth, 2020), references, enumeration symbols, and sentences shorter than three characters.

Implementation Details For both approaches, we measured sentence similarity in terms of the cosine of their embeddings that is generated with Sentence-BERT (Reimers et al., 2019). Recall that our graph-based summarization has three parameters, d_1–d_3, for representativeness, argumentativeness, and contrastiveness respectively. In our experiments, we tested different parameter values between 0.1 to 0.9 with a step size of 0.1 on the *controversial-contexts* dataset. We consider *Contra-PageRank* with $d_3 = 0$ as a baseline, since it disregards contrastiveness. We call it in our experiments here *Arg-PageRank*, but note that this is different than our representative approach in Section 3.1 since it considers similarities between sentences of only the input argument. As for the Comp-Summarizer, to obtain ground-truth generic snippets $\bar{S}$ that are necessary for the supervised training, we consider the argument's conclusion as a proper generic snippet. To this end, we used the *args.me* corpus and heuristically generate generic snippets based on the sentences' overlap with the conclusion using the algorithm of Bista et al. (2020). Similar to Contra-PageRank, we also assess different combinations of values for the hyperparameters, including the contrastiveness weight λ. We used 5-fold cross-validation to evaluate each combination, aiming to minimize the average loss on the data. The optimization worked for 300 epochs with a learning rate of 0.1.

[2] https://www.args.me/api-en.html

[3] https://en.wikipedia.org/wiki/Wikipedia:List_of_controversial_issues

λ	Contrastiveness	Argumentativeness	Representativeness
0.000	0.059	**0.637**	**0.823**
0.500	0.074	0.632	0.803
0.875	**0.086**	0.624	0.720

TABLE 3.5: Automatic evaluation scores of *Comp-Summarizer* for three different values of the contrastiveness weight λ. The best value in each column is marked in bold.

Automatic Evaluation As mentioned, no datasets with ground-truth contrastive snippets exist, and the manual creation of such snippets is arguably arduous. Therefore, we stick to automatic measures that intrinsically assess snippet quality below, in order to evaluate different parameter value combinations and to select some for the manual evaluation. In particular, we capture *contrastiveness* in terms of silhouette analysis score, an intrinsic cluster measure for quantifying clusters quality, as follows. Given a set of snippets $\mathbf{S} = \{S_1, \dots S_k\}$ generated for a set of arguments $\mathbf{A} = \{A_1, \dots A_k\}$, we pseudo-cluster the embedding of all arguments' sentences, with each snippet S_i as one centroid.[4] This way, we can quantify the clusters' quality using silhouette analysis: The more contrastive snippets are, the better the clusters they form, reflected in a higher silhouette score. As for *representativeness*, we compute the mean similarity between the sentences of a snippet S and those of the respective argument A. Finally, we approximate *argumentativeness* by argument quality, employing the BERT model of Gretz et al. (2020b) trained on a regression task to predict the argumentative quality score of a sentence. Here, we implemented the topic-independent version of their approach.

Table 3.4 presents three selected combinations of parameter values that demonstrate the limits of contrastiveness and representativeness for *Contra-PageRank* as well as their trade-off: As expected, setting d_1 to 1 (and, thus, ignoring the other terms) maximizes representativeness, while the best contrastiveness score comes from increasing d_3 to 0.7 (third row). In the second row, we show a value combination that better balances representativeness and contrastiveness. As for argumentativeness, we observed little differences across parameters, which could be the result of the simple lexicon-based method of weighting argumentativeness.

In Table 3.5, we explore the trade-off between representativeness and argumentativeness for *Comp-Summarizer*, showing evaluation scores for selected values of the contrastiveness weight λ. Analogously, a higher λ results in more contrastiveness but less representativeness, while ignoring the contrastiveness term ($\lambda = 0.000$) leads to the best representativeness. A medium value (here, $\lambda = 0.500$) yields a better balance between the three scores.

[4] A snippet's embedding is averaged from its sentences' embeddings.

Approach	Contrastiveness	Representativeness Score	
		Average (± Std.)	Median
Contra-PageRank	***83%**	**3.13 (± 1.15)	3
Comp-Summarizer	*81%	****3.76** (± 1.25)	**4**
Arg-PageRank	65%	3.50 (± 1.35)	**4**

TABLE 3.6: Manual evaluation results for the three compared approaches on a sample of 50 cases: Contrastiveness, in terms of the percentage of generated snippets that were seen most representative of *their* input argument, and representativeness, in terms of the average and median score. Results highlighted with * and ** are significantly better than Arg-PageRank with confidence level of 95% and 90% respectively.

Manual Evaluation To gain more reliable insights into the effectiveness of our approaches in generating contrastive and representative snippets, we conducted a study with four human annotators, who are university students with good English skills. We chose the variants of the two approaches that yielded best contrastiveness above: the third row of Table 3.4 for *Contra-PageRank*, and the third of Table 3.5 for *Comp-Summarizer*. As a baseline focusing on representativeness, we also included the *Contra-PageRank* variant in the first row of Table 3.4. We refer to the latter baseline as *Arg-PageRank*.

For evaluation, we randomly selected 50 samples of three arguments, $\mathbf{A} = \{A_1, A_2, A_3\}$, and we repeated the following process once for each of the three approaches. For each sample, we first generated the respective snippets, $\mathbf{S} = \{S_1, S_2, S_3\}$. For every snippet $S_i \in \mathbf{S}$, two annotators then manually rated how representative S_i is on a 5-point Likert scale, once for each argument in $\mathbf{A}$. We defined representativeness to our annotators by how much the snippet is covering the main gist, thought, or quintessence of the argument.[5] From this, we infer that S_i is contrastive, if it obtained a higher representativeness score for A_i than for all $A_j \neq A_i$. Before doing so, we made one adjustment, though: Since all three approaches are extractive, the annotators would have easily recognized the argument from which S_i was extracted and, consequently, have scored that argument higher. To avoid this bias, we applied automatic rewriting to all snippets using the PEGASUS transformer (Zhang et al., 2020).

Table 3.6 shows each approach's contrastiveness as the percentage of cases where a generated snippet, S_i, got the highest representativeness score for its input argument, A_i. *Contra-PageRank* generated contrastive snippets most often (83%), while *Arg-PageRank* led to contrastive snippets only in 65% of all cases. In other words, 35% of the snippets of *Arg-PageRank* were mistakenly seen as representative of other arguments by the annotators. This result underlines the importance of fostering snippets to be contrastive. The best trade-off is achieved by *Comp-*

[5]For an easy task distribution, we divided the 50 samples into two sets of 25 samples and gave each set to two annotators.

Topic: Cloud Seeding
Argument-1: Cloud seeding should be used worldwide. This is because, according to both EcoHearth.com and Weather Modifications.org , cloud seeding is safe and virtually harmless to the environment. It can safely cause rain in drought-ravaged areas and keep farms from failing. We should institute cloud seeding in areas where it is necessary. **- ArgPageRank's Snippet:** Cloud seeding should be used in certain areas **- Comp-Summarizer's snippet:** Cloud seeding is safe and harmless to the environment according to both EcoHearth.com and Weather Modifications.org **- Contra-PageRank's snippet:** Cloud seeding should be used in certain areas
Argument-2: Thank you, instigator for providing the resolution. I accept all the proposed terms. Comments I'd like to confirm whether the the embryonic dust cloud theory follows as the popular scientific consensus that a planetary system is created from a nebular of ionised gas where denser and more compact regions form the precursors to a planetary system's celestial bodies. I'd also like to ask who coined "Embryonic Dust Cloud Theory" as I don't want to be unintentionally misrepresenting a scientist's work which may slightly differ from the widely accepted theory. **- ArgPageRank's Snippet** I would like to know who came up with the idea of "Embryonic Dust Cloud Theory" as I don't want to be misrepresenting a scientist's work which may slightly differ from the widely accepted theory **- Comp-Summarizer's snippet** I'd like to confirm that the popular scientific consensus is that a planetary system is created from a nebular of ionised gas where denser and more compact regions form the precursors to a planetary system's heavenly bodies **- Contra-PageRank's snippet** I'd like to confirm that the popular scientific consensus is that a planetary system is created from a nebular of ionised gas where denser and more compact regions form the precursors to a planetary system's heavenly bodies.
Argument-3: Since you have failed to give me an example of an instance where another material has been used instead of silver iodide and was successful, i'll have to ignore that argument. You stated yourself it was lethal. It doesn't matter if the chemical is fairly diluted, it is still dangerous and can cause serious harm to ecosystems. The testing of the soil is faulty and unreliable, so it very possible other studies don't have accurate information. In conclusion, cloud seeding should not be used. This is because it is plainly unnatural and has already wreaked havoc on several ecosystems. Silver Iodide is a harmful chemical that should never be used in the first place. Vote Con! Thanks for the good debate. **- ArgPageRank's Snippet:** There will be no new evidence or arguments to be formed during this round. **- Comp-Summarizer's snippet:** Since you didn't give me an example of an instance where another material was used instead of silver iodide, I'll have to ignore that argument. **- Contra-PageRank's snippet:** Cloud seeding should not be used because the chemical is still dangerous and can cause serious harm to the environment.

TABLE 3.7: Example arguments on *Cloud Seeding* along with the snippet generated for each by our two approaches and the baseline

Summarizer which generated the most representative snippets while maintaining contrastiveness almost as often as *Contra-PageRank* (81%). The average inter-annotator agreement of the two annotator pairs was substantial, 0.74 in terms Krippendorff's α, suggesting reliable results.

Example analysis In Table 3.7, we present three arguments on the topic *Cloud Seeding*, along with the snippets generated by each of the approaches. These snippets are the paraphrased version of the top two sentences selected from the argument. We notice that the baseline *Arg-PageRank* tends to select general sentences like *"Cloud seeding should be used in certain areas."* or *"no new evidence or arguments to be found.."*, while *Comp-Summarizer* generated snippets that focus on aspects unique to the argument like *scientific consensus"* and *"harmless to the environment"*.

Follow-up Study on Key Point Modeling

In this follow-up study, we explore new emerging techniques and their effect on key point modeling. On the one hand, we experiment with new methods for assessing the argumentativeness of sentences by integrating them into our Representativeness Approach and testing the effectiveness gained on the argument snippet generation task. On the other hand, we explore the potential of large language models (LLMs), namely ChatGPT, in generating argument snippets.

Argumentativeness Recall that our modeling of key points rely on a scoring function arg to estimate the argumentative quality of each sentence in the argument. We explore two other argumentative scoring techniques and their effect on the snippet generation task – again as a proxy for the task of discussion space identification. In particular, we experiment with our Representative Approach (Arg-PageRank) by computing the argumentativeness score arg using the argumentative quality approach of Gretz et al. (2020b) (*Argument Quality*) and the claim identification approach of Chakrabarty et al. (2019a) (*Claim Identifier*), and comparing them to the lexicon-based scoring method initially used in our approach (*Lexicon*). Additionally, we assess the effectiveness of these scoring methods in generating argument snippets independently from the Arg-PageRank algorithm by forming a snippet from the top two scoring sentences according to the evaluated method. We carry the evaluation on the *expert-representative-snippets* test set from Subsection 3.3.1.

ChatGPT As mentioned, most recently, the NLP field has witnessed great success for large language models (LLMs), such as ChatGPT, across many NLP tasks. Therefore, we give insights into the effectiveness of ChatGPT in generating snippets for arguments. In particular, for each argument arg in our *expert-representative-snippets* dataset, we prompt ChatGPT as follows: *Select two sentences that best*

Approach	ROUGE-1	ROUGE-2	ROUGE-L	ACCURACY
Claim-Identifier	0.47	0.35	0.38	0.33
Arg-LexRank (Claim-Identifier)	0.50	0.40	0.43	0.39
Argument Quality	0.51	0.40	0.43	0.39
Arg-LexRank (Argument Quality)	0.52	0.42	0.43	0.43
Lexicon	0.45	0.34	0.39	0.32
Arg-LexRank (Lexicon)	**0.55**	**0.46**	**0.48**	**0.44**
ChatGPT	0.49	**0.46**	**0.48**	-

TABLE 3.8: The ROUGE and accuracy scores between the generated snippet and the ground-truth ones for the different argumentativeness scoring methods arg independently and as part of the Arg-LexRank algorithm as well as for ChatGPT.

represent the following argument: "<arg>*"*, where <arg> refers to the argument's text. We collect Chat GPT output and evaluate it against the ground-truth snippets compared to our approach and the baselines.

Results Besides the accuracy computed as before, Table 3.8 also presents the ROUGE scores of predicting the ground-truth snippets computed based on the similarity with what is being selected as a snippet by the approach and the ground-truth snippets. As for generating snippets by ranking the argumentativeness of sentences, the best-performing approach is the Argument Quality of Gretz et al. (2020b) with an accuracy of 0.39 compared to 0.32 and 0.33 for the Lexicon-based and Claim-Identifier approaches. Nevertheless, when integrating these argumentative approaches as scorers arg in the Arg-LexRank, the best boost comes rather from the Lexicon approach resulting in the best accuracy of 0.44. Finally, we see that simply prompting ChatGPT to generate a snippet for arguments reaches the effectiveness of our approach in terms of ROUGE-2 and ROUGE-L. However, our approach has the advantage of being open-source and interpretable.

From Snippets to Key Points

We evaluate our approach to modeling key points on the snippet generation task. If our approach can extract sentences that form a helpful snippet, then these sentences can also form representative key points. In this task, we took the two top-scoring sentences from each argument to form a snippet, and we compared these snippets to other snippets generated by various baselines. We empirically found that our approaches generate better snippets compared to baselines. In the following, we will explore the effectiveness of our approach in aggregating all the scored sentences into one final key point set, given the scores generated by our key point modeling approach.

3.3.2 Key Point Analysis

Recently, Bar-Haim et al. (2020a) introduced the Key Point Analysis (KPA) task, which compromises two complementary subtasks: (1) *generating key points from a given set of arguments* and (2) *matching these key points to the input arguments*. Since this task is very similar to ours, especially the first subtask, we use it as a proxy to evaluate our approach. As mentioned, due to the chronological development of our research, the key point modeling component models only the representativeness and argumentativeness of sentences, and the key point aggregation component ensures key point diversity by applying the algorithm mentioned in Section 3.2. We leave the evaluation of modeling contrastiveness for future experiments. In the following, we will first describe the task and data made available to study it. We will then give details on how we implement our approach to address this task and discuss the automatic and manual evaluation results of our participation in the Shared Task version of the KPA task (Friedman et al., 2021b).

Task Description

In the context of computational argumentation, Bar-Haim et al. (2020a) introduced the notion of a *key point* as a high-level argument that resembles a natural language summary of a collection of more descriptive arguments. Specifically, the authors defined a good key point as being "general enough to match a significant portion of the arguments, yet informative enough to make a useful summary." In this context, the KPA shared task consists of two subtasks as described below:

1. *Key point generation* Given a set of arguments on a certain topic that are grouped by their stance, generate five to ten key points summarizing the arguments.
2. *Key point matching.* Given a set of arguments on a certain topic that are grouped by their stance and a set of key points, assign each argument to a key point.

This definition aligns with our discussion space identification task, with an extra requirement here to perform matching between the extracted discussion points and arguments in the input collection. Therefore, we test our approach on this task.

Data

We start from the dataset provided by the organizers as described in Friedman et al. (2021b). The dataset contains 28 controversial topics, with 6515 arguments and a total of 243 key points. For each argument, its stance towards the topic is given. Each topic is represented by at least three key points, with at least one key point per stance and at least three arguments matched to a key point. From

the given arguments, 4.7% are unmatched, 67.5% belong to a single key point, and 5.0% belong to multiple key points. The remaining 22.8% of the arguments have ambiguous labels, meaning that the annotators could not agree on a correct matching to the key points. The final dataset contains 24,093 argument-key point pairs, of which 20.7% are labeled as matching. To develop our approach, we use the split as provided by the organizers with 24 topics for training, four topics for validation, and three topics for testing.

Key Point Generation

We apply our Representative Approach from Subsection 3.1 to this task as follows. We employed Spacy (Honnibal et al., 2020) to split the arguments into sentences and construct an undirected graph with the arguments' sentences as nodes. As a quality check, similar to Bar-Haim et al. (2020b), we filter in only sentences between 5 and 20 tokens that do not start with a pronoun. To model sentence argumentativeness, we experiment with two of the methods introduced in Subsection 3.3.1; *Lexicon-based* and the *Argument Quality* of Gretz et al. (2020b). To model centrality, instead of using semantic similarity scores between sentences, we use contrastive learning to learn from the provided data a similarity model between arguments and key points as explained in Section 3.1. Details on the training process are presented below. We use the trained model to embed sentences and then compute edge weights between pairs of nodes (sentences) using cosine similarity between the embeddings. Additionally, we implemented two thresholds to filter out nodes (sentences) of lower argumentativeness (min_arg) and edges of the lower matching score (min_match).

To find the best hyper-parameters for our approach (damping factor d, min_arg, and min_match) and select the best argumentativeness scoring method (*Lexicon* or *Argument Quality*), we optimize on the validation set towards the best F1-score computed by ROUGE method when considering the longest common sequence matching (ROUGE-L) between the top ten sentences (key points) ranked by our approach and the ground-truth key points. As mentioned, to obtain the final set of key points, we rank the scored sentences in descending order, and we aggregate them by iterating through this ranked list and adding each sentence to the final set if its maximum matching score with the already selected candidates is below 0.8.

Results Table 3.9 shows the ROUGE-L score for some selected hyper-parameters settings of our approach. We observe that the best score is achieved by using the Argument Quality approach of Gretz et al. (2020b) as the argumentativeness scoring method with a damping factor of 0.4, argument quality (min_arg) and matching (min_match) thresholds of 0.8 and 0.2 respectively. Hence, we use these specification in our final approach to generate key points from the test set.

Argument Quality Scoring				Lexicon Scoring			
P	min_arg	min_match	R-L	P	min_arg	min_match	R-L
0.4	0.8	0.2	**0.267**	1.0	0	0.2	**0.248**
0.2	0.2	0.4	0.259	0.8	1.0	0.8	0.226
0.4	1.0	0.8	0.196	0.4	0.8	0.4	0.199

TABLE 3.9: ROUGE-L F1-score (R-L) for different hyper parameter settings of our approach considering the damping factor p, the argumentativeness and matching thresholds (min_arg, and min_match), and the argumentativeness scoring method (Lexicon and Argument Quality (Gretz et al., 2020b))

Model	Rank	Strict mAP	Relaxed mAP	Avg. of mAP (r)	p@50% (r)
Our Approach	1	0.789 (1)	0.927 (4)	0.858 (1)	0.848 (1)
NLP@UIT	2	0.746 (3)	0.930 (3)	0.838 (2)	0.827 (3)
ModrnTalk	3	0.754 (2)	0.902 (6)	0.828 (4)	0.806 (5)
Enigma	4	0.739 (5)	0.928 (4)	0.833 (3)	0.828 (2)

TABLE 3.10: Evaluation results of the task of argument and key point matching. We only show top four ranked approaches. Ranks on each measure are in brackets. Besides the official evaluation metrics, two other measures are considered: the average value of the strict and relaxed mAP values, and p@50% for the strict view. Table taken from Friedman et al. (2021a)

For the final step to eliminate redundancy in the generated key points, we excluded sentences with a matching score higher than 0.8 with the selected candidates.

Key Point Matching

We employed RoBERTa-large (Liu, 2019) for encoding the tokens of the two inputs of key point matching to the siamese neural network, which acts as a mean-pooling layer and projects the encoder outputs (matrix of token embeddings) into a sentence embedding of size 768. We used Sentence-BERT (Reimers and Gurevych, 2019) to train our model for 10 epochs, with batch size 32, and maximum input length of 70, leaving all other parameters to their defaults. In the development phase, we trained our model on the training split and evaluated on the validation split provided by the organizers. For the final submission, we did a five-fold cross validation on the combined data (training and validation splits) creating an ensemble for the matching (as per the mean score).

Shared Task's Evaluation Results

Baselines Besides our approach, 17 models were submitted to the key point matching task and five models to the key point generation task. In the follow-

Approach	Relevant	Representative	Polarity
BarH	2	1	1
Our Approach	2	1	2
Enigma	4	4	2
XLNet	1	3	4

TABLE 3.11: Final evaluation results of key point generation, comparing our approach (Our Approach) to the top two submitted approaches, along with Bar-Haim et al. (2020b) approach (barH). The generated key points were ranked in terms of how relevant and representative (Rep.) of the input arguments, as well as their polarity

ing, we will describe some of these baselines that appear in the top-ranked list in our results table. For key point matching, Team NLP@UIT created an ensemble of five models fine-tuned on different folds of the dataset starting from ALBERT XXLarge (Lan et al., 2019) pre-trained language model. Team ModernTalk fine-tuned RoBERTa-base Liu (2019) model on a concatenated version of the argument and key point pairs. The Enigma team used DeBERTa-Large (Martin et al., 2022) to generate an embedding from a concatenation of key points, arguments, and topics. This embedding is then concatenated with the corresponding POS tags and fed to two more dense layers. As for the key point generation task, the Enigma team dealt with the task as an abstractive summarization task. They fine-tuned Pegasus (Zhang et al., 2020) with argument and topic concatenation as input and the key points as a summary. The XLNet team applies their matching model to all possible pairs of arguments, and arguments with the highest average matching scores were considered the final key points.

Evaluation Measures For key point matching, the organizers computed both strict and relaxed mean Average Precision (mAP) following Friedman et al. (2021b). In cases where there is no majority label, for instance, the relaxed mAP considers them to be a match, while the strict mAP considers them as not matching. Besides these two scores, the precision at 50% (p@50%) in the strict scenario is computed. For the key point generation task, the organizers evaluated the generated key points through a crowdsourcing study in which submitted approaches were ranked according to the quality of their generated key points in terms of relevancy (Relevant), representativeness of the collection (Representative), and whether they correctly reflect the stance of the collection towards the topic (Polarity).

Results In key point matching, our approach obtained a strict mAP of 0.789 and a relaxed mAP of 0.927 on the test set, the best result among all participating approaches (Table 3.10). For key point generation, our approach was ranked top one in terms of generating representative key points. However, in terms of polarity, our approach is ranked second after the baseline of Bar-Haim et al. (2017), indicating

	t_1/con		t_1/pro		t_2/con		t_2/pro		t_3/con		t_3/pro		All	
Approach	**R-2**	**R-L**	**R-2**	**R-L**	**R-2**	**R-L**	**R-2**	**R-L**	**R-2**	**R-L**	**R-2**	**R-L**	**R-2**	**R-L**
ChatGPT	**0.03**	**0.24**	0.06	0.26	**0.11**	0.32	**0.10**	**0.30**	**0.01**	0.10	**0.02**	**0.29**	**0.06**	0.25
Ours	0.02	0.23	**0.12**	**0.31**	0.10	**0.33**	0.03	0.28	0.00	**0.22**	**0.02**	0.26	0.05	**0.27**

TABLE 3.12: The ROUGE scores of the key points generated by ChatGPT and our approach (ours) with and without the filtering step computed for every argument collection that corresponds to a topic t_i and a stance (pro/con), as well as on average (All)

the need for explicit control of the polarity of generated key points. Details of the evaluation can be found in the organizers' report (Friedman et al., 2021b).

Key Point Generation via ChatGPT

In a follow-up experiment, we assess the performance of our approach in extracting key points compared to ChatGPT. To this end, we use the test split provided by the organizers of the key point analysis task (Gretz et al., 2020b), which consists of three topics with a corresponding set of pro and con arguments for each, resulting in 6 argument collections. Each collection has 4 to 7 ground-truth key points.

ChatGPT For each of these argument collections, we prompt ChatGPT as follows:

> We define key points to be:
>
> 1. Key points may be viewed as high-level arguments
> 2. They should be general enough to match a significant portion of the arguments yet informative enough to make a useful summary
> 3. Key points are bullet point summaries of an argument collection
>
> Now, given the following arguments on the topic "<topic>", what are the main 5 key points in them: <args>

In the prompt, <topic> and <args> are placeholders for the corresponding topic and its argument collection.

Results Table 3.12 shows the ROUGE scores of the key points generated by ChatGPT and our approach (ours) computed for every argument collection that corresponds to a topic t_i and a stance (pro/con), as well as on average (All). Regarding the ROUGE scores, we can see no clear conclusion as to which approach works better on the task. Therefore, we present the top three key points generated by each

Topic: (t_1) Routine child vaccinations should be mandatory **- Stance:** Pro	
Ours	(1) Routine child vaccination should be mandatory so that the general population can thrive by being free of deadly diseases (2) Child vaccinations should be mandatory to provide decent health care to all (3) Child vaccinations should be mandatory so our children will be safe and protected
ChatGPT	(1) Routine child vaccinations should be mandatory to prevent the spread of diseases and protect children from deadly illnesses (2) Mandatory vaccinations ensure the health and safety of the general population, including vulnerable individuals such as infants and the elderly (3) Vaccinations save lives and prevent the unnecessary suffering caused by preventable diseases
GT.	(1) Routine child vaccinations are effective (2) Child vaccination saves lives (3) Routine child vaccinations are necessary to protect others
Topic: (t_2) Social media should be regulated by the government **- Stance:** Con	
Ours	(1) Regulation by government of social media platforms would be detrimental to free speech around the world (2) Government regulation of social media would be harmful to democracy (3) Social media platforms should not be regulated by the government because it is an invasion of privacy
ChatGPT	(1) Government regulation of social media platforms would be harmful to democracy and freedom of speech (2) Social media platforms should not be regulated by the government as it would infringe on privacy and freedom of expression (3) Social media platforms are private companies and should not be controlled by the government
GT.	(1) Social media regulation is not effective (2) Social media regulation harms privacy (3) Social media regulation harm freedom of speech and other democratic rights
Topic: (t_3) The USA is a good country to live in **- Stance:** Con	
Ours	(1) The poorest in society don't have access to either good health care or an adequate benefits system (2) The USA is not a good place to live in because of the wide variance between rich and poor (3) The USA is not a good place to live
ChatGPT	(1) High crime rates and lack of safety (2) High tax rates and expensive cost of living (3) Political divisions and social unrest
GT.	(1) The US has unfair health and education policies (2) The US has a problematic/divisive political system (3) The US has high taxation/high costs of living

TABLE 3.13: Three example topics and the corresponding top three key points generated by our approach (Ours) and by ChatGPT, along with the ground-truth (GT.) key points

approach in Table 3.13 to gain more insights. The full table can be found in the Appendix (Table A.1). In all cases, we can observe that both approaches provide

similar coverage of the ground-truth key points. These results indicate that despite the comparable simplicity of our approach to ChatGPT, it can still produce competitive results. Moreover, our approach has the advantage of being interpretable. For example, we can explain why our approach generated a certain key point by pointing out the key point's argumentative and centrality scores in the argument collection.

3.4 Concluding Remarks

In this chapter, we studied the task of identifying the discussion space of a given topic. We considered each sentence in the argument collection as a candidate key point. Therefore, our approach to the task consisted of two steps: modeling the key point candidacy of sentences and then aggregating top candidates to form the final set of key points. To model the key point candidacy, we considered sentences representative of the argument collection and argumentative, and we proposed methods to diversify the collection by modeling contrastiveness as an extra criterion focusing on sentences that highlight the uniqueness of their argument compared to the whole collection. Our experiments demonstrate that the candidacy criteria we considered are important to extract sentences representing the final key points. Nevertheless, potential limitations and future directions are worth highlighting.

Limitations First, we motivated the need to identify a topic's discussion space by its usefulness in guiding the generated arguments of a debate technology. Nevertheless, our work has focused on studying methods to identify such a discussion space without proposing a method for how this discussion space can guide the generation process. Future research might consider methods that incorporate the extracted talking points into the argument generation process by simply using them as extra input along with the topic or as conditions to be satisfied during the decoding stage (Dathathri et al., 2020). Second, we introduced contrastiveness as an extra criterion when modeling the key point candidacy to ensure the diversity of key points. Although we experimentally demonstrated how this criterion leads to extracting snippets from arguments that highlight their uniqueness in the collection, we did not evaluate its effect on the overall task of extracting key points.

Finally, our approach is extractive. We focus on extracting sentences because they are more intuitive and easy to interpret. Nevertheless, the extractive nature of our approach limits their applicability in cases where the reasoning of an argument remains implicit, an apparent phenomenon in argumentation. In these cases, one might use other approaches that infer missing components (Chakrabarty et al., 2021, Alshomary et al., 2020b) or consider abstractive summarization approaches capable of inferring new summaries of arguments.

Contributions In debates, human contenders might use an argument search engine (Wachsmuth et al., 2017b) as one of their sources of information to consume potential arguments on the web. Efficient presentation of retrieved arguments in such search engines is crucial to boost their usability. Generating *snippets* for arguments is one way of boosting this efficiency because it helps users assess the relevancy of the retrieved arguments to their information needs. In this scenario, we envision our approaches to generating effective snippets of arguments as a helpful tool to assess humans in exploring argument search engines. Future research can construct user studies to investigate how useful these snippets are in assessing human debaters.

Overall, our work in this chapter contributes to the overall debate scenario by providing methods to generate main talking points for the discussed topic that can be used to guide the generated arguments in the discussion. We provided empirical evidence of the importance of modeling the sentence's centrality, contrastiveness, and argumentativeness as criteria for their candidacy of being final key points. As highlighted in Figure 1.3, our conceptual approach can make use of these collected key points to ensure the relevancy of generated arguments to the discussion space. The next Chapters will then explore the other two relevant aspects of debates: the audience and the opponent's argument.

Chapter 4

Modeling the Audience

In this chapter, we move our view to study *audience* as another aspect that can influence the effectiveness of synthesized arguments in debates. As discussed in Chapter 1, addressing the audience's belief system is crucial to achieving agreement among audiences (van Eemeren and Houtlosser, 1999). For example, in a debate on *Globalization*, potential topics to be covered can be *Economy*, *Culture*, or *Services*. However, knowing that the audience has the same cultural conservationism values would restrict the selection to *Culture* . Therefore, appropriate usage of presentational devices might put a con argument as follows:

"Globalization has destabilized previously immutable social institutions, shifting cultural value away from old traditions to new more individualistic and market-friendly ideas."

Accordingly, operationalizing this type of knowledge about the audience as a component of debate technologies could benefit the production of arguments that bridge the gap between disputed parties by focusing on shared beliefs rather than divisive ones (Feinberg and Willer, 2015).

Several works studied the persuasiveness of natural language arguments subject to the audience's beliefs (Durmus and Cardie, 2018, El Baff et al., 2020) and found correlations between the two. Others extended argumentation frameworks to consider the audience's values when assessing the strength of an argument (Bench-Capon et al., 2002). Nevertheless, research on argument generation studied this task independent of a target audience. We argue that an argument is more effective when it utilizes knowledge about the beliefs of its audience. Therefore, in our work, we study the research question of *what models can be used to represent the audience's beliefs and how to use them to generate more effective arguments?*

To address this research question, we introduce the audience-based argument generation task that focuses on generating argumentative texts on a topic considering a specific audience representation. To address this task, as highlighted in Figure 4.1, we propose an approach of two components; *modeling beliefs* and *argument generation*. The first component builds a computational model of the audience

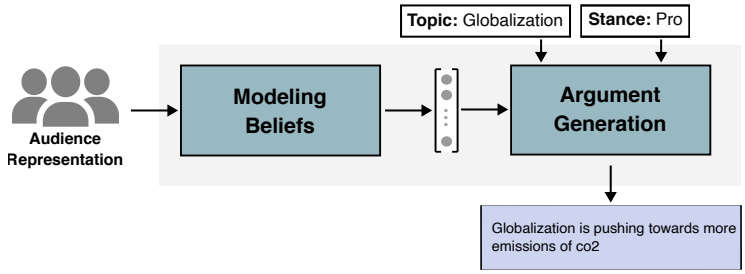

FIGURE 4.1: Our general approach for generating audience-aware arguments. Given an audience representation, first we build a model of their beliefs, then we encode the model along with the topic and stance to generate a final argument that is relevant to the topic and reflecting the audience's beliefs.

given a specific representation, and the second component uses the computational model to control the generated argument to fit the given audience representation.

In particular, we study two representations; the audience's stance on popular topics, called big issues hereafter, and their moral foundations (Haidt and Joseph, 2004). For each representation, we propose a realization of our approach as follows. In the stance-based approach, our modeling belief component takes as an input the stances of the audience on a set of big issues and produces a context vector. This context vector is then used in the argument generation component to tailor the synthesized claim toward the given audience. In the moral-based approach, we model the audience as a vector of five dimensions representing their moral foundations. We develop a model to identify morals in argumentative texts, and we use this classifier along with the vector representation to adapt Project Debater's argument (Slonim et al., 2021) to fit the audience's moral system.

Our experiments evaluate the applicability of encoding the stance-based model an audience into single claims, as well as the effect of generating morally framed arguments on different audiences. Our main findings are the following: (1) Results demonstrate the potentiality of synthesizing claims that reflect specific stances on topics, (2) The stance-based generated claims are more similar to the ground-truth ones written by humans, (3) The morally framed arguments have more effect on the audience than their corresponding generic ones. Overall, our methods to build a computational model of the audience and using it to synthesize corresponding arguments can be used as a component in debate technologies to enable them to achieve better reach to their audience.

In the following, we will first present the modeling belief component of our approach (Section 4.1). We will discuss the two audience representations we considered in this research and how we computationally model them. Next, in Section 4.2, we will present the argument generation component of our approach. We will provide details on how we encode the audience model into the argument generation

process in order to generate arguments targeting specific audiences. Finally, Section 4.3 will list a series of experiments that we perform to evaluate our hypothesis on the applicability and importance of encoding knowledge about the audience into the process of argument generation.

4.1 Modeling Beliefs

In the following, we will present the first component of our approach that deals with building a computational model of the given audience representations, which can be stances on big issues or moral foundations, resulting in two realizations of our approach: stance-based and moral-based. The following section will then discuss integrating these models to generate argumentative texts tailored to the given audience.

4.1.1 Stance-based Approach

Given an audience defined by their stances on a set of big issues, we introduce two ways to model them: as a learned contextual embedding vector or as a bag-of-words reflecting their stances. The following presents details on these two models.

Contextual Embedding As highlighted in Figure 4.2 below, we start with a binary vector $\overrightarrow{u} \in \{0,1\}^k$, where values one and zero reflecting pro and con stance respectively, and k being the number of big issues considered. We then project this vector into a new embedding space via a feed-forward neural network with a learned weight matrix W_U, producing a new vector:

$$\overrightarrow{v} = \sigma(W_U \cdot \overrightarrow{u})$$

We integrate this embedding into a Sequence-to-sequence generation model (Figure 4.2) as a contextual embedding. Given a set of users with their stances on big issues and their claims on specific topics, the model learns the correlation between the input stances and the output claims.

Belief-based Bag-of-Words We build a bag-of-words representing an audience's beliefs from their stances on the big issues. For example, an audience pro *abortion* would likely be pro *choice*. Hence, words such as *right* and *choice* are candidates to be included in their belief-based bag-of-words. To this end, we first build two bag-of-words representations for each big issue, one for the pro and one for the con side. For a specific audience, we then construct a belief-based bag-of-words aggregated from their stances on big issues. As shown in Figure 4.2, we use this vector to control the decoding process of a language model to generate arguments containing words from this vector.

To build a representative pro and con bag-of-words for each big issue, we follow the topic signature approach of Lin and Hovy (2000). Given a big issue, we first collect from some corpus of arguments three sets: relevant pro arguments R_{pro}, relevant con arguments R_{con}, and a random set of non-relevant arguments $\hat{R}$. For each relevant set (R_{pro} and R_{con}), we compute a likelihood ratio for all its words with respect to $\hat{R}$ and keep only words with a score higher than a specific threshold τ, resulting in two sets of words, W_{pro} and W_{con}. Since a word may appear in both sets, we remove it from the set where it occurs fewer times. Finally, we sort words according to their likelihood ratio and keep in both W_{pro} and W_{con} the top k words, forming the final pro and con bag-of-words, respectively. Given an audience (represented by stances on big issues), we construct a belief-based bag-of-words:

$$U_{bow} = W_1 \cup W_2 \cup \ldots \cup W_n$$

where W_i is the pro bag-of-words if the stance is pro and the con bag-of-words otherwise.

4.1.2 Moral-based Approach

Another aspect we take on modeling the audience is through the lens of the moral foundation theory (Haidt, 2012). As detailed in Section 2.2.2, this theory projects the belief system of humans into five foundations: care, fairness, loyalty, authority, and purity. Studies built on this theory demonstrated the correlation between these morals and human judgment. Therefore, we take on studying the feasibility of generating morally framed arguments computationally and the effect of these arguments on different audiences. Given a specific audience, we represent them as a binary vector of five dimensions, where each dimension reflects whether they value the corresponding moral foundation (value of one) or not (value of zero). For example, a user who values care and fairness will be mapped into $\overrightarrow{v} = [1, 1, 0, 0, 0]$, where the first two indices represent care and fairness, respectively, and the last three represent loyalty, authority, and purity.

4.2 Argument Generation

In this section, we discuss the second component of our approach, which integrates the belief model of the audience from the previous section into the argument generation process. As mentioned, we consider two realizations of this component, one for each audience model. For the stance-based approach, we provide two implementations. The first implementation builds on Li et al. (2016), equipping a sequence-to-sequence (Seq2seq) model with the contextual embedding learned in the first component, while the second implementation uses the belief-based bag-of-words to control the output of a pre-trained argumentative language model (LM)

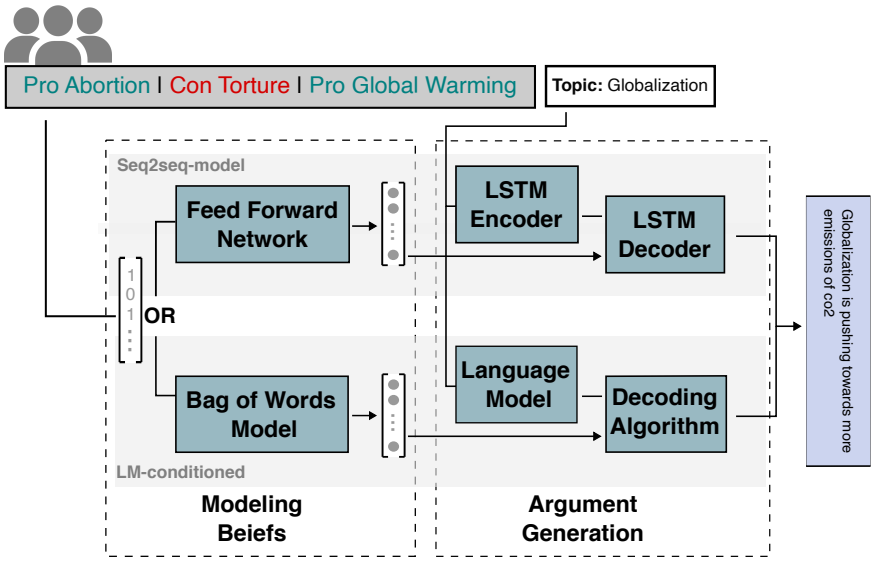

FIGURE 4.2: Our stance-based argument generation approach: We model the audience's stances on big issues via either a feed-forward network that produces a latent vector used as a context to a Seq2seq model or as a bag of words that is used to control the output of an argumentative language model.

using the algorithm of Dathathri et al. (2020) to ensure resembling the beliefs. Both focus on generating single *claims* rather than full arguments to keep it simple and because claims are the main units around which arguments can be constructed. Second, our moral-based realization extends the capabilities of *Project Debater* (Slonim et al., 2021), a hybrid approach of multiple components designed to generate high-quality arguments that compete with human arguments. Building on this technology helps us focus on evaluating the impact of morally framed arguments, as it ensures a certain base quality level in the generation. In the following, we provide a detailed overview of these implementations.

4.2.1 Stance-based Approach

As illustrated in Figure 4.2, both implementations of this approach are based on our previous work (Alshomary et al., 2021a). They first build a computational model of the audience's stances on big issues and then use this model to control the final generated argument.

Seq2seq-based Model

Given a topic, as a sequence of words $T = (w_1, w_2, ..., w_n)$, a contextual embedding of the audience $\overrightarrow{v}$ (presented in previous section), and a claim as a sequence of words $C = (w_1, w_2, ..., w_m)$, first an LSTM-based encoder consumes the input

topic. It produces a hidden state $\overrightarrow{h}$, which initializes the LSTM-based decoder. Following Li et al. (2016), The contextual embedding $\overrightarrow{v}$ learned in Subsection 4.1 serves as the contextual embedding in the model. The difference between the speaker model in Li et al. (2016) and this model is that the vector $\overrightarrow{v}$ is not explicitly predefined but rather learned from the data, while in our model, it is already predefined as a binary vector representing the audience's stances on big issues. In case two audiences have the same stances on big issues, their user vectors are identical, while any two user vectors in Li et al. (2016) will never be identical. By augmenting the Seq2seq model with a context vector, the model is supposed to capture the correlation between the audience's stances on big issues and the corresponding claims. Once the correlation is learned, the model can generate a claim utilizing not only the topic but also the stances on big issues of an audience.

Conditioned Language Model

As mentioned in Section 4.1, this approach first models the audience's stances on big issues as a bag-of-words vector. It then uses the topic as a prompt for a pre-trained argumentative language model (LM) to synthesize a claim conditioned using the algorithm of Dathathri et al. (2020). The synthesis process is illustrated in Figure 4.3. A standard LM is not enough since we aim to generate *claims* in particular. So, to model argumentative language, we take an LM pre-trained on general language and fine-tune it on a large set of arguments (in our experiments, we use the corpus of Ajjour et al. (2019b)). The result is an LM that is able to generate argumentative text.

Given an audience represented via U_{bow} and a topic, we use the topic as a prompt and the user's bag-of-words U_{bow} to condition the generated claim (see Figure 4.3). In particular, given a transformer-based LM (Vaswani et al., 2017), a token x_{t+1} is generated at each time step as follows:

$$o_{t+1}, H_{t+1} = LM(x_t, H_t) \tag{4.1}$$

$$x_{t+1} \sim p_{t+1} = Softmax(W \cdot o_{t+1}) \tag{4.2}$$

where H_t represents the history of the LM. Using the algorithm of Radford et al. (2019), called Plug and Play LM (PPLM), an update to the past, ΔH, is computed to control the generated claim based on the sum of the log-likelihood $p(U_{bow}|x)$ of all words in the belief-based bag-of-words. Then the new history, $\hat{H}_t = H_t + \Delta H_t$, is used as in the previous equations to draw a new distribution $\hat{p}_{t+1}$, of which a new token is sampled. To ensure fluency in the generated text, ΔH is modified to minimize the Kullback–Leibler (KL) divergence between the output distribution of the modified LM and the original one.

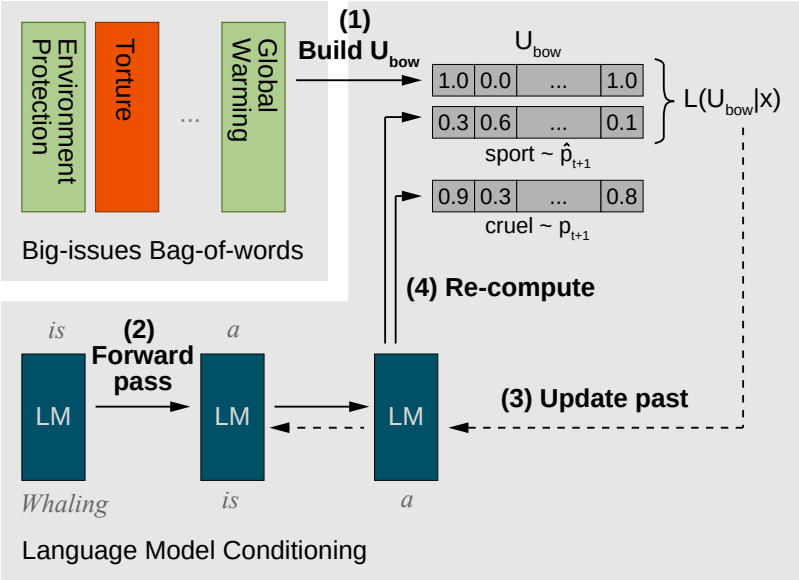

FIGURE 4.3: The synthesis process of the conditioned LM on the topic "Whaling", given a user, defined by stances on a set of big issues (Torture, Environment Protection, etc.), taken from Alshomary et al. (2021a). Steps: (1) Building U_{bow}, based on stances (2) Forward pass through the LM to generate a token, *sport* (3) Updating the LM history H_t, based on $p(U_{bow}|x)$, and (4) Generating from the new history $\hat{H}_t$ a new token *cruel*.

In short, by fine-tuning an LM on argumentative text, we tune it to generate claims. We use the topic as a prompt to ensure the claim is on the topic. Finally, the PPLM algorithm ensures that the beliefs represented as a bag-of-words U_{bow} appear in the claim.

4.2.2 Moral-based Approach

As presented in Section 4.1, in this realization, we model the audience as a vector of five dimensions reflecting their moral beliefs (say, *loyalty*, *authority*, and *purity*). Accordingly, given a topic (say, "globalization"), and the audience's moral vector, our approach retrieves appropriate argumentative texts matching the moral vector. It then constructs an argument based on these texts by extending the Project Debater API. Figure 4.4 shows the high-level process of the proposed system. In this scenario, we consider the stance as an extra input to control whether the generated argument is pro or con to the topic, a feature available in Project Debater's API.

In the following, we will present our approach to identifying morals in texts based on our previous work Alshomary et al. (2022a), which is an essential step to collecting appropriate argumentative texts that fit a specific audience. We then introduce the main components of our Moral-based Approach.

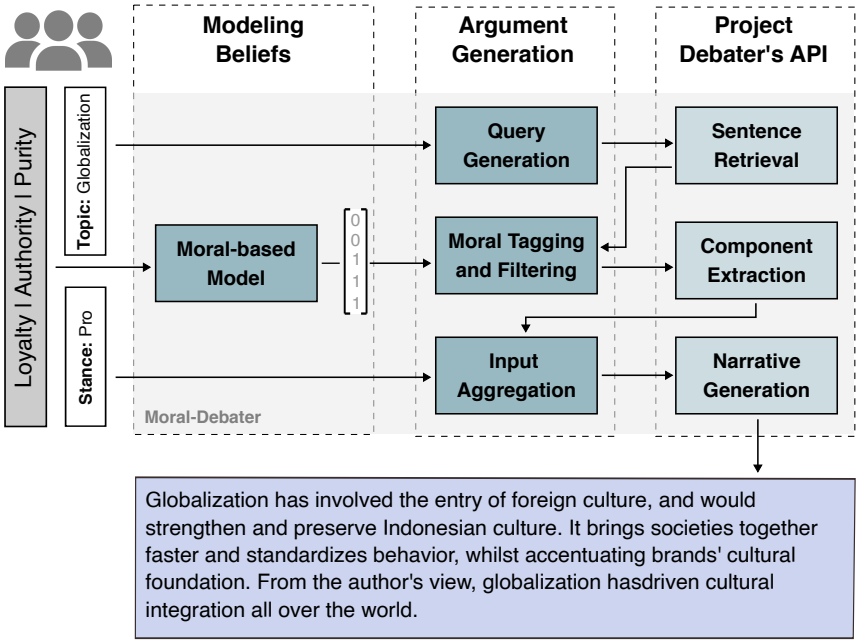

FIGURE 4.4: Our moral-based argument generation approach. The approach extends the capabilities of Project Debater by moral tagging and filtering to output a morally framed argument for a given topic, a stance on the topic, and a set of morals. Given a topic, we issue queries to retrieve argumentative sentences from the sentence retrieval component. Second, these sentences are tagged with morals and filtered according to the moral-based model. Third, we extract argument units (claims/evidence) from the filtered sentences using component extraction. Finally, the argument units are aggregated and sent to the narrative generation component of Project Debater to generate the final argument.

Identifying Moral Foundations in Texts

Existing approaches to mining morals from texts are either lexicon-based or machine learning-based. A number of datasets with morals have been constructed for domains such as social media or news articles. For argumentative texts, Kobbe et al. (2020) manually annotated a small dataset of 220 arguments, which is only suitable for evaluation. We, therefore, decided to develop a moral foundation classifier based on data collected automatically using distant supervision.

Particularly, to circumvent the need for annotated data, we construct a training dataset following a distant-supervision approach. We observe that moral foundations are revealed as aspects of concern in discussions of controversial topics. For example, when discussing *School Uniform* from the *authority* perspective, aspects such as *respect* and *obedience* often arise. Given this observation, our distance supervision approach works as follows. We start from the dataset of Schiller et al. (2020), which contains short argumentative texts on eight topics along with as-

Moral	ML	GC	Ab.	DP	MW	NE	Clo.	SU
Care	14%	31%	19%	13%	16%	32%	20%	10%
Fairness	13%	26%	28%	22%	23%	9%	13%	16%
Loyalty	9%	13%	14%	21%	34%	20%	24%	38%
Authority	54%	25%	21%	7%	8%	2%	25%	8%
Purity	10%	5%	17%	36%	19%	37%	17%	28%

TABLE 4.1: Distribution of the five moral foundations across the eight topics, Marijuana Legalization (ML), Gun Control (GC), Abortion (Ab.), Death Penalty (DP), Minimum Wage (MW), Nuclear Energy (NE), Cloning (Clo.), School Uniform (SU). The topics *Cloning* and *School Uniforms* are used for validation, all others for training.

pects annotated automatically for each text, and the lexicon of Hulpus et al. (2020), which connects moral foundations to Wikipedia concepts. Accordingly, we then assigned each text a set of moral foundations based on the aspects appearing in it. After filtering out arguments without any mapping and balancing the data across the five moral foundations, this resulted in a dataset with 230k argumentative texts and the corresponding morals. We split the dataset into six topics for training and two for validation (testing will happen on other data below). Details on the distribution of the morals across topics are found in Table 4.1.

We rely on a BERT-based classifier to identify morals in texts (Devlin et al., 2018), starting from the pre-trained *bert-based-cased* model. We fine-tuned the model on our training set for three epochs with a batch size of 16 and a learning rate of $3 \cdot e^{-5}$. In the training phase, the input was an argumentative sentence and the corresponding moral foundation. Since an argument may contain multiple sentences, each reflecting a specific moral, an argument's final set of morals consists of all sentences' morals predicted with confidence above 0.5. To assess the classifier's effectiveness more reliably, we trained six models on different random samples of size 50k and computed their average F_1-score.

Main Components

First, given an input topic, the *Query Generation* component retrieves a collection of relevant argumentative sentences from Project Debater's index, which contains 400 million news articles. The articles are split into sentences and indexed along with several meta-annotations. We generate several queries containing only the topic keywords without any topic expansion to focus on relevant sentences. We restrict the retrieved sentences to only those annotated as having sentiment or causality markers. We give more details on the constructed queries in the experiments subsection.

Second, the trained BERT-based classifier, presented earlier, is used in the *Moral Tagging and Filtering* component to annotate each argumentative sentence for all likely moral foundations. It then filters out those sentences that either do

not have any moral or contain at least one moral not given as input. Next, through Project Debater's API, the system generates a likelihood score for each of the remaining sentences, reflecting whether it contains a claim or evidence following the approach of Ein-Dor et al. (2020). We instruct the API to keep only sentences having a claim with a likelihood higher than `claim_threshold` or evidence with a likelihood higher than `evidence_threshold` (the exact thresholds are given below). Additionally, the API identifies claim boundaries for sentences containing claims and extracts the exact span of text containing the claim.

Third, the *Input Aggregation* component aggregates the given list of claims and evidence sentences with the input topic and stance. It then uses Project Debater's narrative generation API to generate the final argument. The narrative generation identifies the stance of claims and evidence towards the topic according to the approach of Bar-Haim et al. (2017). Only those matching the input stance are kept. Redundant elements are then filtered out, and the remaining ones are grouped into thematic clusters, where a theme is a Wikipedia title (Slonim et al., 2005). The process of building these clusters also includes extracting one claim that represents the theme. Each theme will then be represented by a paragraph in the output argument. Finally, a set of algorithms is used to perform various kinds of re-phrasing on the argument level (e.g., pronoun resolution) and on the paragraph level (e.g., ensuring that different arguments are put together) (Slonim et al., 2021).

4.3 Evaluation

In this section, we present a series of experiments to evaluate our hypothesis on the importance of integrating knowledge about the audience in argument generation. The first two Subsections 4.3.1 and 4.3.2 detail our experiments to automatically and manually evaluate the applicability of encoding the audience's beliefs, represented as stances on big issues, into generated claims. We then move our view to the moral representation of an audience. Subsection 4.3.3 presents the results of our distance supervision approach to identify morals in argumentative texts. Finally, Subsection 4.3.4 provides insights into our evaluation study of the effect of morally-framed arguments on different audiences.

4.3.1 Automatic Evaluation of the Stance-based Models

We aim to evaluate if considering the audience's opinions on big issues, represented as stances, helps generate claims closer to the ground-truth claims and reflect the input opinions on big issues. We first explain the evaluation metrics and data preprocessing we used in our experiments. We then describe the implementation details of the methods and baselines we compared. Finally, we analyze the results and findings of our automatic evaluation.

Dataset	# Claims	# Topics	# Users
Training set	41 288	22 241	5 189
Validation set	5 028	2 450	2 509
Test set	5 154	2 728	2 512
Full dataset	51 470	27 419	5 189

TABLE 4.2: Number of claims, topics, and users in each of the training, validation, and test sets.

Data

To train our models, we need a dataset with information about audiences revealing their beliefs and their arguments on various topics. Here, we build upon the dataset introduced by Durmus and Cardie (2018), which they collected from *debate.org*. Users can debate controversial topics and share their profiles on this online platform. We consider each user here equivalent to a specific audience. The dataset contains users' arguments as answers to topic questions and various user information, including a user's self-specified stances (pro or con) on up to 48 predefined popular controversial topics. We consider these topics the big issues on which we build the audience's belief system. In our dataset, for the task at hand, we keep only users with at least three arguments and state their stance on at least one of the big issues. For those, we collected their arguments along with the topics and stances. The dataset contains around 51k claims on 27k topics from 5k users. We randomly split the dataset per topic into 10% test and 90% training split. We use 10% of the latter as the validation set. Statistics are given in Table 4.2.

Since this dataset contains arguments rather than single claims, we preprocess this data by automatically extracting argumentative claims from each argument. In particular, we apply the claim detection approach of Chakrabarty et al. (2019b) by scoring the likelihood of each sentence being a claim and only keeping the one with the highest score as the user's claim on the topic. To evaluate the model, we created a sample of 100 arguments, and two annotators decided whether the extracted sentence represented a claim on the given topic. In terms of full agreement, the model extracted claims correctly in 81% of the cases, the Cohen's κ inter-annotator agreement being 0.3. We note that this preprocessing step produces some noise in the data, mainly affecting the training of our Seq2seq model below.

Evaluation Measures

On one hand, we compute the BLEU and METEOR scores of the generated claims with respect to the ground-truth claims. On the other hand, we compute the likelihood that the generated claims possess textual features reflecting the user's beliefs by computing the accuracy of a classifier in predicting the user's stances on big

Approach	BLEU-1	BLEU-3	METEOR
S2S-baseline	18.2%	0.44%	**16%**
S2S-model	*__18.4%__	*__0.46%__	**16%**
LM-baseline	9.6%	**0.26%**	8%
LM-conditioned	*__12.0%__	0.16%	*__11%__

TABLE 4.3: BLEU and METEOR scores of the claims of each evaluated approach compared to the ground-truth claims. Values marked with * are significantly better than the respective baseline at $p < .05$ (student's t-test).

issues when trained on the generated claims. We compute this accuracy for each of the big issues individually and report the results for all of them. In particular, we perform the following three steps for an evaluated model. First, we generate claims for all given users and topics in the test dataset. Second, we keep only instances in which users have a stance (pro/con) on the tested big issue, and we split the filtered dataset into training and testing. Finally, we train a simple TF-IDF-based linear classifier on the training set to predict the stance on the big issue given the text of the claim. The accuracy of the classifier on the test split then quantifies the likelihood of the generated claims possessing textual features that reflect the stance on the corresponding big issue.

Model Implementation

We implement the Seq2seq-based model based on the OpenNMT framework (Klein et al., 2017). The encoder and decoder of the model are each two-layer LSTMs of hidden size 512 with GloVe word embeddings of size 300. Users' stances on big issues are represented as a one-hot encoded vector and then projected into 16 dimensions space through a one-layer dense neural network. We train the model with the Adagrad optimizer and refer to it as *S2S-model*.

For the conditional language model, we constructed the pro/con relevant argument sets (R_{pro}, R_{con}) by querying the respective big issue from the API provided by Ajjour et al. (2019b) and extracting pro/con arguments from the top 60 results. We used the same corpus for the non-relevant argument set ($\hat{R}$) and randomly selected 100 arguments. We eliminated all words with a score under $\tau = 10$ and finally kept the top $k = 25$ words from each set (R_{pro}, R_{con}) to represent the bag-of-words. We refrained from tuning the parameters here since we lack ground truth. To model the argumentative language, we fine-tuned the GPT-2 model on the corpus of Ajjour et al. (2019b), which contains around 400k arguments. We perform the fine-tuning using the transformers framework (Wolf et al., 2019). We used the topic as a prompt to trigger the generation process. However, since some topics are phrased as a question (e.g, "Is abortion wrong?"), we extracted the noun phrase from the topic and used it as a prompt. We used the PPLM implementation

Approach	Ab.	DP.	GM.	DL.	GW.	EP.	MM.	SB.	MW.	BF.	ALL
Ground-truth	0.49	0.59	0.55	0.55	0.55	0.55	0.50	0.53	0.48	0.62	0.52
LM-baseline	0.48	0.50	0.54	0.49	0.54	0.56	0.51	0.45	0.59	0.46	0.50
LM-cond.	*__0.58__	*0.53	0.45	__0.56__	*__0.61__	__0.58__	__0.58__	__0.53__	__0.65__	0.50	__0.54__
# Training	1 610	1 532	2 098	1 538	1 960	2 196	2 096	1 370	1 580	1 092	-
# Test	350	366	196	316	156	86	138	294	172	280	-

TABLE 4.4: Accuracy of each classifier trained on claims generated by the evaluated approaches to predict the stance, on the 10 most frequent big issues (AB. = Abortion, DP.= Death Penalty, GM.= Gay Marriage, DL.= Drug Legalization, GW.= Global Warming, EP.= Environment Protection, MM.= Medical Marijuana, SB.= Smoke Ban, MW.=Minimum Wage, BF.= Border Fence) as well as on average over all 48 big issues. Values marked with * are significantly better than corresponding baseline at $p < 0.05$ according to a one-tailed Student's t-test.

(Dathathri et al., 2020)[1] for conditioning the generated claim. We call this model the *LM-conditioned*.

Baselines We compare our two approaches to the corresponding version without stances on big issues as input to evaluate the gain of encoding users' beliefs. We refer to these baselines as *S2S-baseline* and *LM-baseline*, respectively.

Results

Table 4.3 shows the results of our approaches and the baselines in terms of BLEU and METEOR. For *S2S-model*, the BLEU scores of our approach are significantly better than the baseline. The *LM-conditioned* is significantly better than the baseline version in terms of BLEU-1 and METEOR. In general, the S2S-model has the highest scores across all measures. The reason may be that it was trained in a supervised manner on the given dataset, whereas the *LM-model* was only fine-tuned in an unsupervised way on a different argument corpus.

Regarding the encoding of user stances, Table 4.4 shows the accuracy of a linear classifier trained to predict the stance from the claims generated by each approach as well as from the ground-truth, on average and on the ten most frequent big issues. The full table for all the 48 big issues is in the Appendix (Table A.3). The best average accuracy across all the big issues is achieved by the *LM-model* (0.54). By comparison to the corresponding baselines, the LM-model and the *S2S-model* generated claims that boosted the accuracy of the stance classifier on 33 (69%) and 21 (44%) of all big issues, respectively. Overall, in 20 of the big issues, the best accuracy was achieved on the claims generated by the conditioned LM, compared to only nine big issues for the S2S-model. These results indicate that

[1] step-size=0.15 and the repetition-penalty=1.2

the LM-conditioned can better encode a user's beliefs, modeled as stances on big issues, into generated claims.

4.3.2 Manual Evaluation of the Stance-based Models

To obtain more insights into belief-based claim generation, we let users manually evaluate the output of the given approaches. Upon inspecting a sample of generated claims by our approaches, we noticed that the *LM-conditioned* produces more fluent and informative texts. Accordingly, we focused on the LM-conditioned and its baseline in the evaluation, where we conducted two user studies. The goal of the first was to assess the quality of the big-issue bag-of-words collected automatically, and the second targeted the output of the LM-model, its baseline, and a variant that utilizes a manually refined bag-of-words.

Automatic Collection of Bag-of-words

We evaluated only the top ten big issues to keep the manual annotation effort manageable. Two authors of our work (Alshomary et al., 2021a) categorized each word in the pro/con bag-of-words of the corresponding big issue into five categories, *c1–c5*:

- c1: Word irrelevant to the big issue.
- c2: Relevant word, wrong stance.
- c3: Relevant word, both stances possible.
- c4: Relevant word, correct stance.
- c5: Very relevant word, correct stance.

Table 4.5 shows examples of each relevance category between the collected words and the *Abortion* topic. In order to compute inter-annotator agreement, three big issues were annotated by both annotators, resulting in Cohen's κ of 0.45, reflecting moderate agreement. Afterward, only one annotator continued the annotations for the other big issues. Table 4.6 shows the distribution of words over categories, averaged across the ten big issues. For the bag-of-words representing the pro stance, around 40%

Claim Generation

We evaluate the effectiveness in terms of whether a given generated claim reveals the stance of the given user on a specific big issue as well as how informative the claim is regarding the given topic. Since not all topics are directly related to the big issues that can be revealed in the generated claims, we manually annotated the relatedness of the top frequent 200 topics in the test dataset to the ten most frequent

Stance	c1	c2	c3	c4&c5
Pro	goes, home-made	murder, alive, illegal	aborted, pro-cedures	option, fetus, mother, right
Con	getting, doubles	–	delivery, rate, abort	contraception, conception, sanctity

TABLE 4.5: Example words found in the pro and con relevant arguments of *Abortion* as a big issue, and our manual classification of their relevancy. **c1** is for words that are irrelevant, **c2** are relevant words but wrong stance, **c3** is for relevant words that fit any stance, **c4** is for relevant words and correct stance, and finally **c5** for very relevant words that are in the correct stance

	Irrelevant	Relevant			Very Relevant
Words	**c1**	**c2**	**c3**	**c4**	**c5**
Pro	14%	10%	36%	34%	6%
Con	36%	2%	34%	26%	2%

TABLE 4.6: Distribution of the pro/con bag-of-words, averaged across the top-10 big issues, over the five considered categories: c2 means wrong stance, c3 words that fit both stances, and c3 and c4 represent correct stance.

	Overall			Relatedness L4			Relatedness L3			Relatedness L2		
Approach	**True**	**False**	**Und.**	**True**	**False**	**Und.**	**True**	**False**	**Und.**	**True**	**False**	**Und.**
LM-base.	44%	34%	**22%**	50%	50%	**0%**	55%	31%	14%	**27%**	20%	**53%**
LM-cond.	37%	32%	31%	35%	38%	27%	59%	41%	**0%**	13%	**13%**	74%
LM-man.	**45%**	**26%**	28%	**50%**	**31%**	19%	**61%**	**28%**	11%	25%	18%	56%
GT.	42%	30%	28%	38%	42%	19%	64%	27%	9%	27%	19%	54%

TABLE 4.7: Manual Evaluation: Percentage of cases for each approach (LM-base.= LM-baseline, LM-cond.= LM-conditioned, LM-man.= LM-conditioned (manual)) where the majority of annotators predicted the stance of a generated claim on the given big issue correctly (true), incorrectly (false), or could not decide it (Und.). The overall scores and those for each topic/big-issue relation level are listed.

big issues and created the evaluation sample accordingly. In particular, two authors of our work scored the relatedness of each pair of topics and big issues on a scale from 1 to 4:

4: Topic and big issue are the same. Example: *"gay marriage should be legalized" and "gay marriage"*.

Approach	Overall	Level 4	Level 3	Level 2
LM-baseline	1.8	**2.5**	1.9	1.4
LM-conditioned	**2.1**	2.3	**2.5**	**1.5**
LM-cond. (manual)	2.0	2.3	2.2	1.5
Ground Truth	2.0	1.9	1.8	2.2

TABLE 4.8: Manual Evaluation: Mean informativeness of the claims generated by each approach with regard to the topic (1–3, higher is better). The overall scores and those for each topic/big-issue relation level are listed.

3: A stance on the topic likely affects the stance on the big issue. Example: *"killing domestic abusers" and "death penalty"* .

2: A stance on the topic may affect the stance on the big issue. Example: *"morality" and "abortion"*.

1: Topic and big issue are not related. Example: *"do aliens exist?" and "abortion"*.

Example topics and big issues and their relatedness level is shown in the Appendix (A.2). The two annotators had a Cohen's κ agreement of 0.54. Around 97.4% of all pairs got score 1, 1.1% score 2, 0.8% score 3, and 0.7% score 4. The small percentage of cases that can be evaluated reflects a limitation in the designed evaluation study. Nevertheless, it still allows us to evaluate the effectiveness of our approach for different levels of relatedness. Given the annotated pairs, we randomly selected ten pairs from levels 2, 3, and 4, and we collected all claims on the topic for each pair from the test set, where the author specifies a stance on the corresponding big issue. We randomly select 30 claims each, resulting in an evaluation sample of 90 instances.

We used the crowdsourcing platform *MTurk*[2] for evaluation. For each instance, we showed a topic, a claim, and the corresponding big issue to three annotators. The annotators had to perform two tasks: (1) to predict the stance of the user on the corresponding big issue from the text of the claim, and (2) to rate the claim's informativeness regarding the topic on a scale from 1 to 3.

Table 4.7 shows the percentage of cases in which the majority of annotators predicted the stance correctly (true), incorrectly (false), or could not decide about the stance (und.) from the generated claim. Across the whole sample (Overall), the claims generated by *LM-conditioned (manual)*, the model conditioned on the refined bag-of-words, most often allowed to predict the stance correctly (45%). We thus attribute the low effectiveness of the *LM-model* to the noise generated by the automatic collection of big issues' bag-of-words, especially seeing that the effectiveness improves across all levels when eliminating this noise.

[2]A crowdsourcing platform: `https://www.mturk.com/`

(a) *Topic:* whaling
Big issue: (Pro) Environmental Protection *Level:* 3

Approach	Claim	Stance
LM-cond.	whaling is a very dangerous and cruel practice. It causes great harm to the environment, animals that are caught in it have their lives cut short by being eaten alive.	Pro
LM-baseline	whaling is a sport. It is a sport because it involves a lot of physical exertion and skill.	Undec.

(b) *Topic:* should murder be legalized?
Big issue: (Pro) Death Penalty *Level:* 2

Approach	Claim	Stance
LM-cond.	should murder be legalized? I think that abortion should not be legal. Abortion is killing a baby, and it's wrong to kill an innocent human being.	Undec.
LM-baseline	should murder be legalized? I think so. I think that it should be legalized because it is a good thing	Undec.

TABLE 4.9: A selection of claims generated by the different evaluated approaches for the different association levels between topic and big issue discussed in the text.

Analyzing each relatedness level yields more insights. For relatedness level 4, where the topic is the same as the big issue, the *LM-conditioned (manual)* generated claims where the majority of the cases with known stances were correct (63%). In level 3, we observe that both versions of our approach outperform the baseline in producing claims that express the correct stance on the corresponding big issue with percentages of 59% and 68%, respectively. Finally, at relation level 2, which represents a weak relation between topics and big issues, predicting the stance seems to become hard, as indicated by high percentages of undecided cases. We believe that the weak relatedness made the annotators guess the stance in some cases, leading to unreliable annotations.

Table 4.8 shows the average score of all approaches regarding the informativeness of the generated claims. Here, both versions of our approach achieved better scores than the baseline, matching the ground-truth score. We believe that the low scores of the ground-truth claims stem from the noise generated in the claim detection step. However, the *LM-conditioned (manual)* has a lower score than the *LM-conditioned*. We believe that better encoding of stances on big issues into claims comes with the cost of reducing the informativeness regarding the topic.

Error Analysis Table 4.9 shows some cases from our evaluation. Case (a) shows a working example of which our approach correctly generated a claim on *whaling* from an environmental perspective when conditioned as such. Case (b) is a level 2

	Care			Fairness			Loyalty			Authority			Purity		
Approach	**Pre**	**Rec**	**F_1**	**Pre**	**Rec**	**F_1**	**Pre**	**Rec**	**F_1**	**Pre**	**Rec**	**F_1**	**Pre**	**Rec**	**F_1**
Lexicon	0.64	**0.88**	**0.60**	0.07	**0.70**	0.13	0.09	**0.86**	0.17	0.14	0.63	0.23	0.16	**0.72**	0.27
mBERT	**0.74**	0.38	0.50	**0.47**	0.35	**0.40**	**0.50**	0.10	0.16	**0.43**	0.09	0.14	**0.56**	0.13	0.21
Ours	0.54	0.56	0.52	0.31	0.55	0.37	0.21	0.54	**0.28**	0.23	**0.74**	**0.34**	0.46	0.48	**0.46**

TABLE 4.10: Moral foundation classification: Precision (Pre), recall (Rec), and F_1-score (F_1) of our approach and the baselines for each moral foundation. The best value in each column is marked bold.

example, indicating a limitation in our evaluation, namely, the generated claim reveals a stance on abortion, but we asked about the death penalty.

4.3.3 Moral Identification in Argumentative Texts

We now evaluate the effectiveness of our moral identification model explained in Subsection 4.2.2. First, to assess the quality of the distantly supervised dataset, two authors of the paper manually evaluated the correctness of the assigned morals on a sample of 100 examples. 77% of the cases were considered correct by at least one author, 44% by both. The Cohen's κ agreement was 0.32, which is not high, but in line with other subjective argument-related annotations (El Baff et al., 2018).

Next, to assess the effectiveness of our approach, we consider two baselines. The first is the model performing best in the experiments of Kobbe et al. (2020), a multi-label BERT-based model trained on the Twitter moral corpus of Hoover et al. (2020). We trained our version on the same dataset and referred to it as *mBERT*. The second baseline is a simple lexicon-based approach that computes the frequency of words belonging to each moral foundation (Araque et al., 2020), called *Lexicon* below.[3].

We tested all models on the dataset of Kobbe et al. (2020), which consists of 220 arguments annotated for moral foundations by two annotators. Table 4.10 shows the F_1-score of all evaluated models for each moral as well as the macro F_1-score. Additionally, we show the precision and recall for each approach. In terms of F1-score, our approach outperforms both baselines across three of the five moral foundations as well as on average. We observe that effectiveness varies regarding precision and recall between the Lexicon and the mBERT baseline. The stable effectiveness of our approach across the five morals signals the advantage of the proposed dataset that we used in our approach. Hence, our approach uses this model later for morally framed argument generation. Table 4.12 shows two example arguments with the manually annotated morals and those predicted by the baselines and our approach. We see that the Lexicon baseline assigns all morals

[3]Link: `https://github.com/oaraque/moral-foundations`

Approach	Macro Precision	Macro Recall	Macro F_1
Lexicon	0.18	**0.76**	0.28
mBERT	**0.54**	0.21	0.28
Ours	0.35	0.57	**0.40**

TABLE 4.11: Moral foundation classification: The Macro Precision, Recall, and F_1-score of our approach and the baselines over all the moral foundations classes. The best value in each column is marked bold.

to each argument most of the time, leading to high recall across all morals. The first row of the table shows an example argument from the test set in which our approach detected its *authority* moral while mBERT failed. In the second row, our approach missed the *care* moral in the argument but highlighted *loyalty*, a moral that probably emerges from the aspect of helping each other.

4.3.4 Studying the Effect of Moral Framing

To evaluate our hypothesis on the effect of morally framed arguments, we conducted a user study with two opposing target audiences, *liberals* and *conservatives*. Our primary goal was to investigate whether morally framed arguments are more effective than uncontrolled ones. Additionally, we sought to determine whether differently-framed arguments affect liberals and conservatives differently. In the following, we report on this study. The following subsection will present the study results.

Experimental Setup

We again considered the top frequent ten *big issues* from the website debate.org. For each topic, we used our system to construct three arguments: one argument focusing on care and fairness (*individualizing*), one focusing on loyalty, author-

Argument	GT.	Lexicon	mBERT	Our Approach
This is just wrong we should not insult who we believe in we do not need to know what you people think.	*Authority*	*Care, fairness, authority, purity*	–	*Authority*
Christianity does offer hope in the world. Christianity does tell others to help the poor.	*Care*	*Care, fairness, loyalty, authority, purity*	*Care*	*Authority, loyalty*

TABLE 4.12: Ground-truth (GT.) moral foundations of two example arguments from the dataset of Kobbe et al. (2020) in comparison to the morals assigned by the two classification baselines (Lexicon, mBERT) and by our approach.

.

ity, and purity (*binding*), and one baseline argument where we did not control the morals targeted (*uncontrolled*). We created arguments separately for both stances (pro and con), resulting in a total of $10 \cdot 3 \cdot 2 = 60$ arguments. To construct each argument using our Moral-based model (Subsection 4.2.2), we perform the following. We built four queries, retrieving 10k sentences with 6 to 60 tokens per query. The first query retrieved sentences containing the topic. The second and third queries targeted claim-like sentences, requiring the occurrence of (a) at least one causality marker or (b) both causality and a sentiment marker. Each needed to appear together with the topic in a window of 12 tokens. The last query aimed to retrieve evidence by filtering only those sentences that contained any of the following tokens: "surveys", "analyses", "researches", "reports", "research", and "survey". A moral was assigned to a retrieved sentence if the probability of our classifier was higher than 0.5. After initial tests, we set the `claim_threshold` and `evidence_threshold` to 0.8 and 0.6, respectively. We left all other settings to the default values of Project Debater's API.

Internal Study on Argument Quality

Before we launched our main study, two authors of this work manually assessed the quality of the generated arguments and the morals addressed in each. In particular, each of them read all 60 arguments and ranked their *relevance*, *coherence*, and *argumentativeness* on a 5-point Likert scale. While reading each argument, they also highlighted text spans that they found to reflect a specific moral. Table 4.16 presents the quality scores for each argument type, and Table 4.17 the distribution of moral foundations. Comparing the scores of binding and individualizing arguments to the uncontrolled ones, we see that our method did not notably worsen the quality of the generated arguments. The moral foundation distribution indicates that binding arguments have a relatively higher focus on loyalty, authority, and purity than individualizing arguments and a lower focus on fairness and care. These results support the impact of our method on controlling morals in arguments. Example arguments with and without controlled morals are shown in Table 4.13, and full examples can be found in the Appendix (Table A.6)

External Study on Argument Effectiveness

To answer our research questions, we conducted a two-phase user study on the platform *Upwork*: First, we determined the political ideology of each participant, and then, we let selected participants rank the different arguments.

In the first phase, we asked people living in the US who are experienced in writing and content editing to perform the *Political Typology Quiz*, available through

Topic: Globalization
Binding argument: The crowd raised four issues, explaining its views. The first claim is that globalization is reducing the importance of nation-states. The next issue will show how Globalization and structural forces aggravate poverty. In addition, we will hear about pollution and Culture. ... Lastly, Culture. Globalization has destabilized previously immutable social institutions, shifting cultural value away from old traditions to new more individualistic and market friendly ideas. It is often said to have a negative effect on the world's cultural diversity. Cultural and geographical dimensions of transformational leadership become blurred as globalization renders ethnically specific collectivist and individualistic effects of organizational behavior obsolete in a more diversified workplace.
Individualizing argument: The crowd raised four issues, explaining its views. The first claim is that Globalization on its own cannot end gender inequality. In addition, we will hear about harm, economy and processes. Starting with gender inequality. There are various studies available that depict globalization as a hindrance toward gender inequality. Globalization on its own cannot end gender inequality. Turning to harm. $\cdots$ Globalization is a threat to culture and religion, and it harms indigenous people groups while multinational corporations profit from it. It has been criticized for benefiting those who are already large and in power at the risk and growing vulnerability of the countries' indigenous population. $\cdots$
Uncontrolled argument: The crowd raised four issues, explaining its views. The first claim is that globalisation creates economic and cultural imbalances in developing nations. The next issue will show how globalization is reducing the importance of nation-states. And the third point is that globalization is a threat. In addition, we will hear about processes. Starting with economy. Globalization does not work for all the economies that it affects, and that it does not always deliver the economic growth that is expected of it. Globalisation and neoliberalism have exacerbated already unequal economic relations. Although globalization takes similar steps in most countries, scholars such as Hodge claim that it might not be effective to certain countries and that globalization has actually moved some countries backward instead of developing them. ...

TABLE 4.13: Example generated arguments against *Globalization* for different focused morals. The '$\cdots$' indicates an omitted content due to space limitation.

the Pew Research Center, in order to identify their political ideology.[4] In 17 questions, the quiz asks participants to state their views on controversial issues in the US. The test results place the participants on a spectrum of ideologies from *solid liberal* (left) to *core conservative* (right).

[4]Political Typology Quiz: `https://www.pewresearch.org/politics/quiz/political-typology/`

Ideology	Morals	Rank 1	Rank 2	Rank 3	Mean
Liberals	Binding	23%	37%	40%	2.17
	Individualizing	40%	40%	20%	*1.80
	Uncontrolled	37%	23%	40%	2.03
Conservatives	Binding	25%	27%	48%	2.23
	Individualizing	50%	37%	13%	**1.63
	Uncontrolled	25%	37%	38%	2.13
All	Binding	24%	32%	44%	2.20
	Individualizing	45%	38%	17%	**1.72
	Uncontrolled	31%	30%	39%	2.08

TABLE 4.14: Rank distribution and the mean rank for each type of moral framing (binding, individualizing, uncontrolled) reflecting the effectiveness according to the different participant groups (liberals, conservatives, all). Values marked with * and ** are significantly better than Uncontrolled ranks at $p < 0.1$ and $p < 0.05$ respectively.

In the second phase, we chose only six participants from the first phase due to budget constraints, three solid liberals (one male, two female) and three core conservatives (two males, one female). We showed each of them three arguments (one individualizing, one binding, one uncontrolled) for all 20 topic-stance pairs. The participants read the three arguments for each pair and ranked them by perceived *effectiveness*. We followed El Baff et al. (2018), defining the effectiveness of an argument either by how empowering it is (if the participant has the same stance on the topic) or by how challenging it is (otherwise). For this purpose, the participants self-assessed their stances on each topic on a 5-point Likert scale, from 1 (strongly disagree) to 5 (strongly support) before reading the arguments.[5]

Results

Empowering vs. Challenging Figure 4.5 shows the distribution of challenging and empowering arguments. Liberals were more decisive with their stance on the given topics, with 73% being on the pro side, whereas only 30% of the conservatives were on that side (50% con side, 20% no stance). Since we presented arguments for both sides for each topic, we had an equal distribution of empowering and challenging arguments for the liberals. However, for conservatives, we had 40% empowering and 40% challenging arguments due to the 20% undecided cases. Since arguments supporting one side of a debate are rather challenging for the undecided audience, in our analysis below, we consider the 20% undecided cases to be challenging.

[5]Given an estimated workload of 3 to 3.5 hours, we paid each participant a fixed rate of $75.

Ideology	Moral	Empowering	Challenging	Both
Liberals	Binding	2.27	2.07	2.17
	Individualizing	**1.83**	***1.77**	***1.80**
	Uncontrolled	1.90	2.17	2.03
Conservatives	Binding	2.29	2.19	2.23
	Individualizing	**1.71**	***1.58**	****1.63**
	Uncontrolled	2.00	2.22	2.13
All	Binding	2.29	2.19	2.20
	Individualizing	**1.71**	****1.58**	****1.72**
	Uncontrolled	2.00	2.22	2.08

TABLE 4.15: The mean rank of each type of moral framing (binding, individualizing, uncontrolled) according to the different participants (liberals, conservatives, all) for challenging arguments (opposite stance to participant), empowering arguments (same stance), and both. Values marked with * and ** are significantly better than Uncontrolled ranks at $p < 0.1$ and $p < 0.05$ respectively.

Effectiveness of Moral Arguments Table 4.14 shows the rank distribution for morally-framed arguments (*binding* and *individualizing*) compared to the *uncontrolled* ones for liberals, conservatives, and all together. In general, the participants ranked the arguments framed in terms of fairness and care (*individualizing*) significantly better than the *uncontrolled* ones, with an average rank of 1.72 compared to 2.08.[6] This difference is significant at $p < 0.05$ using the student t-test. This signals a positive answer to our first research question: A focus on morals can make arguments more effective. A closer look at the distribution of arguments at *Rank 1* shows that conservatives were more susceptible to moral arguments (75% binding and individualizing) compared to liberals (63%).

Next, we examine whether arguments with different morals affect liberals and conservatives differently by looking at the achieved ranks of both empowering and challenging arguments.

The argument's effectiveness depending on ideology Looking at the mean ranks assigned by liberals in Table 4.15, we observe that challenging arguments that focus on individualizing morals (care and fairness) are most effective. We validate that the difference is significant for $p < 0.1$ using the student t-test. This is in line with Feinberg and Willer (2015), who found that arguments framed in terms of liberal morals were more convincing to liberals. Notably, this effectiveness slightly decreases when arguments are empowering. A reasonable hypothesis is that, in the case of empowering arguments, the audience may be more interested in the opposing views, which uncontrolled arguments might cover. We investigate this hypothesis further via a follow-up questionnaire below.

[6]The difference in mean ranks is 0.36, 95% CI [0.17, 0.57].

Type	Argumentativeness	Relevance	Coherence
Binding arguments	3.8	3.8	**4.0**
Individualizing arguments	**4.2**	4.0	3.9
Uncontrolled arguments	4.1	**4.1**	3.9

TABLE 4.16: Mean quality scores of the three types of evaluated arguments on a 5-point scale (higher is better).

Type	Care	Fairness	Loyalty	Authority	Purity
Binding	16%	17%	10%	47%	9%
Individualizing	17%	36%	6%	35%	6%
Uncontrolled	11%	21%	4%	54%	10%

TABLE 4.17: Distribution of the five moral foundations found in the three types of evaluated arguments.

Now, we look at the conservatives. Although they also valued the individual arguments the most, we observe that when arguments challenged their views, a focus on binding morals (loyalty, authority, and purity) became slightly more effective than the uncontrolled arguments. Generally, morally framed arguments that challenged the views of conservatives were significantly more effective than uncontrolled ones at $p < 0.1$ using the student t-test.

Agreement across Ideologies We measured inter-annotator agreement between the participants using Kendall's W (Kendall and Smith, 1939). The agreement of all six participants was 0.29. In contrast, when considering liberals and conservatives separately, it increased to 0.35 for liberals and 0.51 for conservatives. This indicates higher agreement between participants having similar political ideologies and matches the common notion that conservatives are more unified in their views than liberals.

Reasons behind Effectiveness Judgments In a follow-up questionnaire, we investigated our participants' judgments. We asked them to self-assess whether they prefer (1) arguments with *knowledge* they have or are not familiar with, (2) arguments that matched or challenged their *own views*, (3) arguments that convince *others* who share or oppose their *views*, and (4) what affected the judgments of argument *effectiveness* more: knowledge or views, each in empowering and challenging cases.

Table 4.18 shows that the participants ranked knowledge as the most relevant *effectiveness* aspect. In terms of *others' views*, the majority valued arguments that focus on the opposing views, whereas preferences differ for empowering and challenging arguments on *own views*. Due to the reliability issues of self-assessment

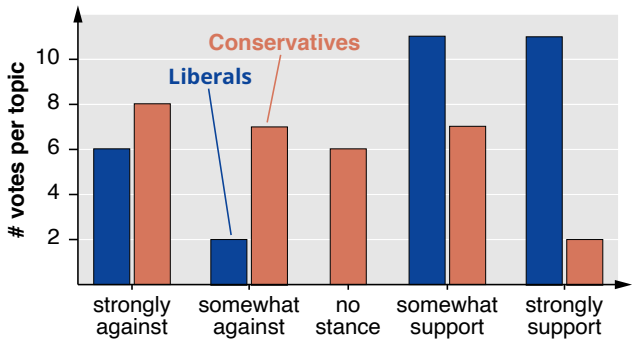

FIGURE 4.5: The distribution of the stances of liberals and conservatives on the ten given topics, on a 5-point Likert scale from 1 (strongly against) to 5 (strongly support) (Alshomary et al., 2022a).

		Empowering	Challenging	All
Knowledge	Know about	33.3%	0.0%	16.7%
	Not familiar	**66.7%**	**83.3%**	**75.0%**
	Neither	0.0%	16.7%	8.3%
Own views	Matched	**50.0%**	16.7%	33.3%
	Challenging	33.3%	**50.0%**	**41.7%**
	Neither	16.7%	33.37%	25.0%
Others' views	Share view	16.7%	0.0%	8.3%
	Oppose view	**66.7%**	**66.7%**	**66.7%**
	Neither	16.7%	33.3%	25.0%
Effectiveness	Knowledge	**83.3%**	**66.7%**	**75.0%**
	Views	16.7%	33.3%	25.0%
	Neither	0.0%	0.0%	0.0%

TABLE 4.18: Distribution of preferences (options) selected by the annotators for each of the four asked questions for empowering and challenging cases.

of one's moral judgments (Pizarro, 2000), we acknowledge the limitation of this study, though. We present details on the questionnaire and its results in the appendix (Subsection A.2.3)

To summarize, our results suggest that when arguments challenge the stance, the morally framed ones are generally more effective, especially for the conservative audience. We find that liberals value arguments that focus on their own morals (care and fairness) the most, especially when their stance is challenged. Although conservatives also value respective arguments, a focus on typically conservative morals (loyalty, authority, and purity) becomes more relevant to them when their stance is challenged. Despite the limited size of our user study, these findings hint

at the importance of utilizing morals to craft more effective arguments in debate technologies.

4.4 Concluding Remarks

This chapter discussed the importance of audience for debate technology to generate more effective arguments. We studied two audience representations and how to model and integrate them into argument generation. In the following, we will iterate over the potential limitations concerning our work and the contributions to the overall picture of effective debate technology.

Limitations First, studying methods to generate arguments targeting a specific audience has a societal impact. Using the audience's beliefs to achieve persuasiveness might be misused for a manipulation attempt. However, we argue that changing people's minds is considered manipulation only if concealed. Therefore, future work should ensure that the mechanics of any AI technology designed to target an audience should be communicated transparently. For example, it should be clear what information technology collects about the user and how it uses it to generate arguments. Due to the need to build a model of the audience, which requires collecting information about them, privacy concerns might arise. General frameworks to protect users' privacy, such as the European Union's General Data Protection Regulation (GDPR)[7], must be maintained. Sousa and Kern (2023) thoroughly surveys different deep learning methods concerned with handling privacy-preserving issues.

Second, our suggested models of the audience are reductionist since they represent the rich human condition as a model of five moral foundations or stances on a set of big issues. These models and the assumptions learned around them only hold for some people. One might identify as a liberal but still have conservative stances on specific issues. Therefore, future work should consider removing these proxy models or relaxing the associated assumptions, for example, learning some latent representation of an audience directly from their texts or other information. However, such an approach might need help with interpretability issues since one can not explain why a particular audience was targeted with a specific argument.

Third, our approaches' ability to generate effective and relevant arguments is limited in two ways. On the one hand, since our stance-based models generate texts from scratch, they inherit challenges such as generating faithful texts. Nevertheless, we focused on generating single claims rather than full arguments to simplify the process of controlling the validity of the generated claims. One can integrate our model into a more extensive system that can perform claim verification (Bekoulis et al., 2021) and support this claim by generating the reasoning around this

[7] https://gdpr.eu/

single claim to get into a final argument. On the other hand, our moral-based model is limited by the coverage of Project Debater's index of argumentative sentences. Content pertaining to more specific topics might not be available, limiting our approach's ability to generate relevant arguments.

Fourth, we integrated the stance-based representation only into the generation models (Seq2seq, and LM-conditioned). Nevertheless, future work should also consider integrating these models into Project Debater API, similar to the moral representation, to have a more consistent assessment.

Finally, in the overall debate technology framework, the argument generation component is envisioned to synthesize arguments adjusted toward the audience and relevant to the discussion space identified in the first component (Chapter 3). However, our implementation focused on only controlling for the audience. Future work can study how to fuse both the audience and the discussion space models into one model that controls the generation of the final argument.

Contributions Overall, we contributed to the field of computational argumentation by introducing the first definition of belief-based argument generation task. We argued for the importance of considering the audience's belief when synthesizing arguments and studied two ways of representing the audiences: their moral foundations and stances on big issues. Our stance-based model builds a vocabulary representation of the audience that can be plugged into any language model to align its generated arguments with the audience's beliefs without retraining them. Our experiments highlight the applicability of generating claims discussing a specific topic while reflecting certain beliefs defined as stances on big issues. To generate morally framed arguments targeting a specific audience, we developed a moral classifier that achieves state-of-the-art results on the task and integrated it into Project Debater. In a user study with liberal and conservative audiences, we empirically show that our approach synthesizes more effective arguments for these audiences.

In the context of debate technology, our proposed methods form a component that builds a belief model of the audience that the debate technology can use to synthesize arguments toward a specific audience, maximizing agreement among them. This component can also be extended to consider the opponent's belief system. Hence generating arguments that are also more effective on the opponent.

Next, we will move our view to the third and last aspect that we address in debate technologies, namely the opponent's argument, where we study methods that analyze the opponent's argument in order to counter it effectively.

Chapter 5

Modeling the Opponent's Argument

An essential building block of debate technology is the counter-argument generation. It is how a debate technology can challenge the opponent by countering their argument and bringing new aspects to enrich the discussion. Nevertheless, generating effective and relevant counters to the opponent's argument is complex. According to Walton et al. (2008), human debaters counter their opponent's argument through one of the following forms; (1) attacking one of its weak premises (undermining), (2) directly refuting its conclusion (rebuttal), or (3) criticizing the reasoning between the premises and the conclusion (undercutting). Therefore, one needs to analyze the input argument to extract knowledge about the main point(s) it implies or its weak premises to synthesize a counter for it successfully. So far, as mentioned in Chapter 1, research in this field did not utilize such knowledge about arguments to improve the generated counters. Therefore, in this chapter, we study the main research question of *how to analyze the opponent's argument and use this knowledge to generate more effective and relevant counters.*

In general, as highlighted in Figure 5.1, we propose an approach of two components; *argument analysis* and *counter generation*. The first component analyzes the opponent's argument to infer relevant information, while the second integrates this information into transformer-based language models to generate the final counter. We study two kinds of information relevant to the counter-argument generation task; *argument conclusion* and *weak premises*. In the following, we discuss two realizations of our approach; *argument-rebuttal* that uses the information about the conclusion to generate a counter, and *argument-undermining* that counters the argument by attacking weak premises analyzed in the opponent's argument.

In the context of the argument-rebuttal approach, since conclusions are only sometimes explicitly stated in the input argument (Al-Khatib et al., 2016), we first start by discussing the conclusion inference task (Section 5.1). To this end, we model the conclusion as a concise statement with a stance toward a specific target and focus on only inferring the conclusion target due to the limitation of generation methods at the time. Later, we build on the advances of pre-trained language models to learn to generate conclusions from data as follows. We use

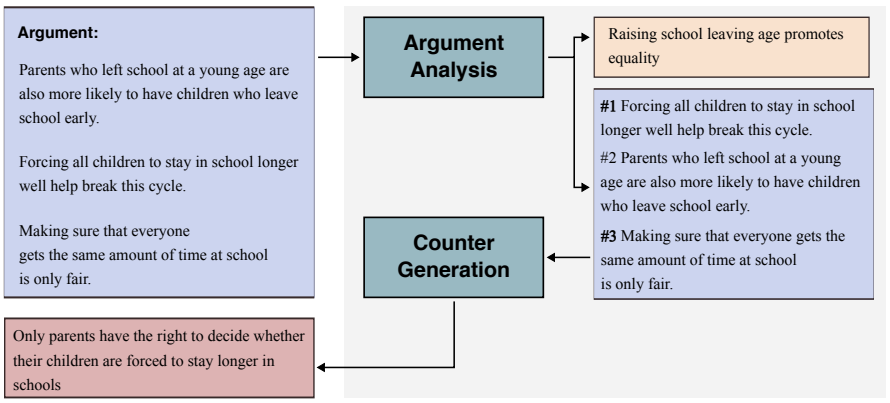

FIGURE 5.1: Our general approach to generating effective counter-arguments has two main components; *Argument Analysis* and *Counter Generation*. In argument analysis, relevant knowledge to the task of counter-argument generation from the opponent's argument is inferred, such as the argument conclusion and weak premises. In the counter generation, this knowledge is then used to generate a counter that undermines the argument by attacking one of the weak premises (as shown in this figure) or rebutting the inferred conclusion.

a multitask approach on top of language models to jointly model conclusion and counter-argument generation tasks (Subsection 5.2.2). Second, we move our view to identify weak premises in the opponent's argument as part of the *argument-undermining* approach. In Subsection 5.1.2, we introduce our previous approach from Alshomary et al. (2021c) that learns to rank premises based on their attackability relevant to their conclusion.

Next, we study how to integrate the extracted knowledge about the conclusion and weak premises into generation models. For this, the second component of our approach employs pre-trained transformer-based models as the backbone. We explore two different modes of integration, pipeline and joint learning. On the one hand, our argument-undermining approach represents a pipeline integration where we take the information about weak premises from the argument analysis component and encode it on the token level to learn to synthesize an appropriate counter (Subsection 5.2.1). On the other hand, as mentioned earlier, in Subsection 5.2.2, we introduce our second approach, argument-rebuttal, that starts from an input argument to generate a conclusion along with the counter in a multitask fashion (Alshomary and Wachsmuth, 2023).

Finally, Section 5.3 delivers a series of experiments to evaluate our approaches and hypotheses. We first verify that our hypothesis pertained to learning the relation between premise and conclusion targets to generate more accurate conclusions. Moreover, we empirically show that transformer-based language models can better learn to generate conclusions if this task is learned jointly with other tasks like counter-argument generation. To analyze weak premises, we demonstrate the

importance of modeling the attackability of premises as a ranking task by achieving state-of-the-art results. Altogether, our experiments demonstrated that inferring the conclusion and weak premises of the opponent's argument can boost the effectiveness of generated counters. Therefore, we argue that integrating our models into debate technologies will increase their overall effectiveness in engaging in debates.

5.1 Argument Analysis

In this section, we present our perspective on what can be useful information about the argument structure that can help synthesize more relevant counters. We specifically consider *the argument conclusion* and *weak premises*. We discuss approaches from our previous work to extract this knowledge from natural language arguments. The following section will then present our two approaches to integrate each of these relevant information into the generated counters.

5.1.1 Conclusion Inference

The conclusion (or claim) of a natural language argument conveys a pro or con stance towards some *target*, such as a controversial concept or statement (Bar-Haim et al., 2017). It is inferred from a set of premises. Conclusions are key to understanding arguments and critical to generating relevant and successful counters. As mentioned, the task of *identifying* conclusions has been studied intensively in the context of argument mining (Stab and Gurevych, 2014b) and automatic essay assessment (Falakmasir et al., 2014). In genres other than essays, however, conclusions often remain implicit since they are clear from the context of a discussion (Habernal and Gurevych, 2015) or hidden on purpose for rhetorical reasons, as is often the case in news editorials (Al-Khatib et al., 2016). Therefore, identifying the conclusion becomes an *inference* task: Given an argument's premises, generate its conclusion.

To address this task, we first present a conceptual approach based on our previous work (Alshomary et al., 2020b) that decomposes the task into three steps (Figure 5.2); (1) conclusion target inference, (2) stance inference, and (3) generating the conclusion's text that carries the inferred stance towards the target. Due to the computational limitation at that time, we focused our research on developing methods to infer the conclusion target. Nevertheless, with the advances of pre-trained language models, we revisit this task by utilizing these pre-trained language models in a multi-task setting (Alshomary and Wachsmuth, 2023). In the following, we will provide details on the first approach and then discuss the second study of utilizing pre-trained language models for conclusion inference.

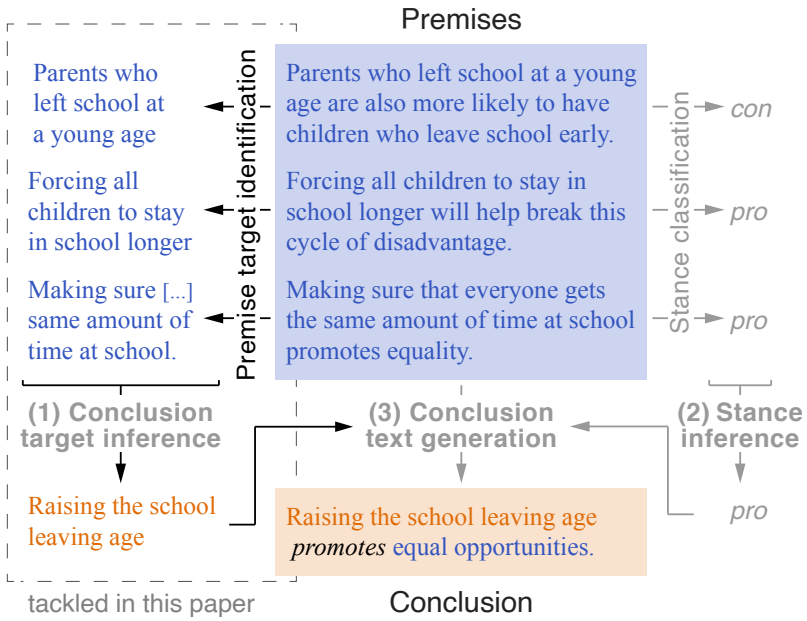

FIGURE 5.2: Illustration of our conceptual approach to generating an argument's conclusion from its premises (Alshomary et al., 2020b). Three conceptual steps are needed. First, the conclusion inference from the argument's premises. Second, inferring the stance of the argument toward this target. Finally, generating the final conclusion text that carries the inferred stance towards the inferred target.

Target-based Conclusion Inference

As sketched in Figure 5.2, we hypothesize that the conclusion target is related to the targets of the argument's premises. To obtain premise targets, we train a state-of-the-art sequence labeling model (Akbik et al., 2018) on target-annotated claims (Bar-Haim et al., 2017). Since the exact relation between premise and conclusion targets is unknown, we develop two complementary inference approaches: One approach ranks premise targets based on their likelihood of being a conclusion target. The other one employs a triplet neural network (Hoffer and Ailon, 2015) that generates a conclusion target embedding from the premise targets in a learned embedding space. It then integrates this network with a knowledge base of targets from which a pre-defined target whose embedding is closest to the generated embedding is chosen. The following provides a detailed overview of our approaches and insights from our experiments to evaluate our hypothesis on the importance of premise targets in the process of conclusion inference.

Premise Target Identification We first identify the premises' targets. Bar-Haim et al. (2017) have introduced the task of identifying these target phrases in an argumentative text. We here tackle it as BIO sequence labeling, classifying each token

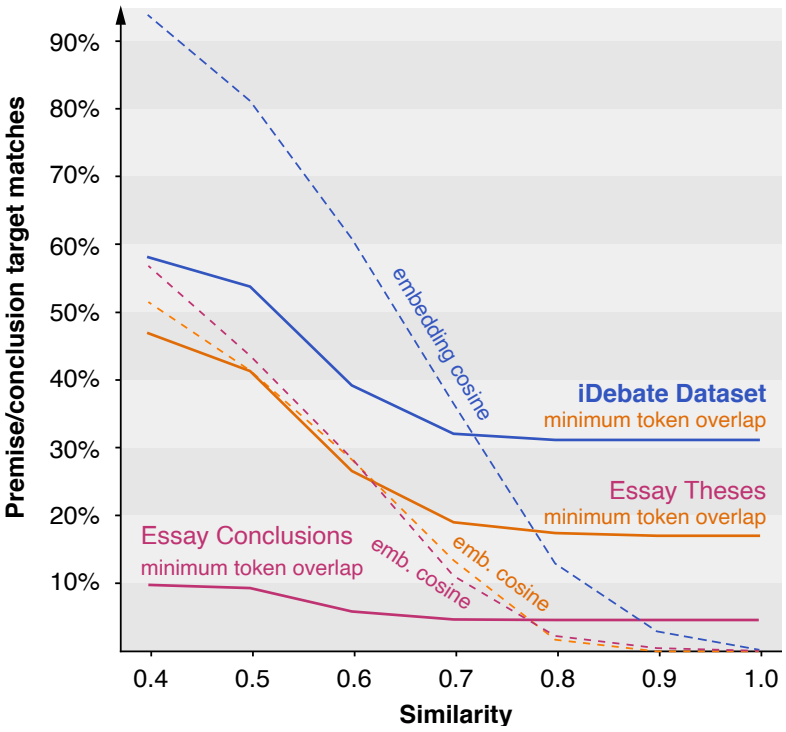

FIGURE 5.3: Percentage of training arguments in the given datasets where the conclusion target matches any of the premise targets, assuming a match when either a certain *minimum token overlap* (solid lines) or some *embedding cosine* similarity (dashed lines) is given. Illustration is taken from Alshomary et al. (2020b).

as being the beginning, inside, or outside of a target. Since premise target identification is not our main focus, we train a state-of-the-art neural sequence tagger (Akbik et al., 2018) on the claim stance dataset of Bar-Haim et al. (2017) and then use it to annotate targets in all input premises automatically.

Inference by Premise Target Ranking A reasonable hypothesis is that one of the premise targets of an argument represents an adequate conclusion target. Therefore, we first simplify the given task into selecting the premise target that most likely represents the conclusion target. Since there is no training data that reflects this likelihood, we follow the idea of importance sampling of Wang and Ling (2016): Given the output of our target identifier on a training instance, we use the percentage of content tokens overlapping between premise targets and the conclusion target as a representativeness label (quantified as Jaccard distance) to construct our training data. Then, we learn a ranking model to predict the representativeness of a candidate premise target based on four features:

1. The average *cosine similarity* of the candidate to the other candidates,

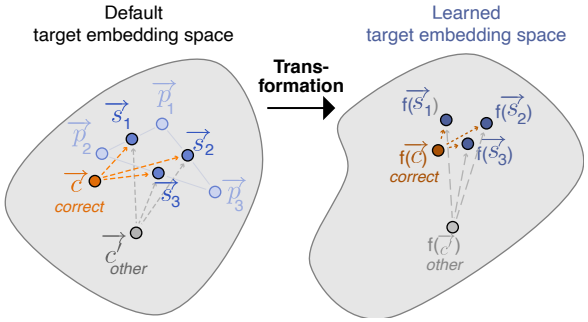

FIGURE 5.4: Sketch of the target embedding space transformation. The distance from the averages $\overrightarrow{s_1}, \overrightarrow{s_2}, \ldots$ of the premise targets to the correct conclusion target $\overrightarrow{c}$ is minimized, the distance to other targets $\overrightarrow{c'}$ maximized. The illustration is taken from Alshomary et al. (2020a).

2. the *number of words* in the candidate,
3. the relative start and end character *position of the candidate* in the covering premise, and
4. the *number of sentiment words* (positive, negative, and neutral) in that premise.

The input of the ranking model is premise targets grouped by argument. During training, a probability is learned to reflect the ordering between each pair of premise targets in an argument with respect to conclusion target representativeness. Then, the model utilizes a cross-entropy loss function to minimize the difference between the learned and the desired probability.

The effectiveness of this approach is limited by the percentage of cases where the conclusion target actually matches any premise target. For a rough estimation, Figure 5.3 shows, based on two different similarity measures, how often at least one premise target matches the conclusion target in the three given training sets. Naturally, it is unclear in general how high the similarity needs to be for actual semantic equivalence.

Inference by Target Embedding Learning To overcome the outlined shortcoming of being restricted to premise targets, we investigate a second hypothesis: An adequate conclusion target can be found in other arguments. To this end, we integrate a neural model with a knowledge base of targets. In particular, our second sub-approach tackles the given task by producing candidate conclusion target embeddings from the (top-ranked) premise targets and then picking the target from a knowledge base whose embedding is most similar to the candidates. In principle, the knowledge base can be built from any corpus of argumentative texts based on our target identifier. In our experiments, we use all conclusion targets extracted from the training split of the datasets.

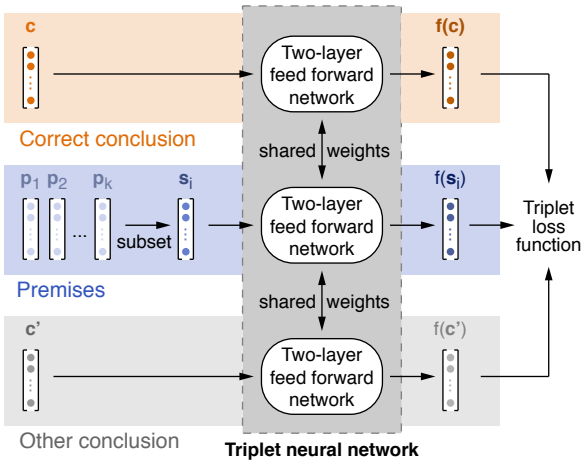

FIGURE 5.5: Our approach to learning conclusion target embeddings. The triplet neural network makes the average embedding **s** of a subset of the premise targets similar to the correct embedding **c**, and dissimilar to others. Illustration is taken from Alshomary et al. (2020b).

To predict a conclusion target embedding, we first get the top $k > 1$ premise targets using our ranking approach and create average embeddings $\overrightarrow{s_1}, \overrightarrow{s_2}, \ldots$ of all $\binom{k}{m}$ possible subsets of these targets with $m > 1$. Then, we learn a function f on training arguments that maps each $\overrightarrow{s_i}$ to a transformed embedding space where it resembles the correct conclusion target $\overrightarrow{c}$ and differs more from other targets $\overrightarrow{c}'$. Figure 5.4 sketches this idea. The best k and m are found by tuning in validation.

As depicted in Figure 5.5, we model f as a *triplet neural network* (Hoffer and Ailon, 2015) with three vectors as an input: an anchor $\overrightarrow{s_i}$, a positive $\overrightarrow{c}$, and a negative $\overrightarrow{c}'$, where $\overrightarrow{c}'$ is a randomly sampled target from the target knowledge base. During training, we create $\binom{k}{m}$ triplets from each argument. Based on these, we utilize the following triplet loss function to minimize the cosine distance d between $\overrightarrow{s_i}$ and $\overrightarrow{c}$, and to maximize d between $\overrightarrow{s_i}$ and $\overrightarrow{c}'$:

$$\max\left\{d\big(f(\overrightarrow{s_i}), f(\overrightarrow{c})\big) - d\big(f(\overrightarrow{s_i}), f(\overrightarrow{c}')\big) + d_{max}, 0\right\}$$

Here, d_{max} represents the maximum distance to be considered, also determined during validation.

During prediction, we employ the trained network to map the average embeddings $\overrightarrow{s_1}, \overrightarrow{s_2}, \ldots$ of all premise target subsets to the transformed embedding space, and compute the average $avg(f(\overrightarrow{s_i}))$ of all mapped embeddings $f(\overrightarrow{s_i})$. Then, we pick the conclusion target $\overrightarrow{c}$ from the knowledge base whose mapped embedding $f(\overrightarrow{c})$ has the minimum cosine distance to $avg(f(\overrightarrow{s_i}))$. This way, we ensure we always end up with a meaningful target. Figure 5.6 sketches the conclusion target inference on the left and exemplifies it on the right.

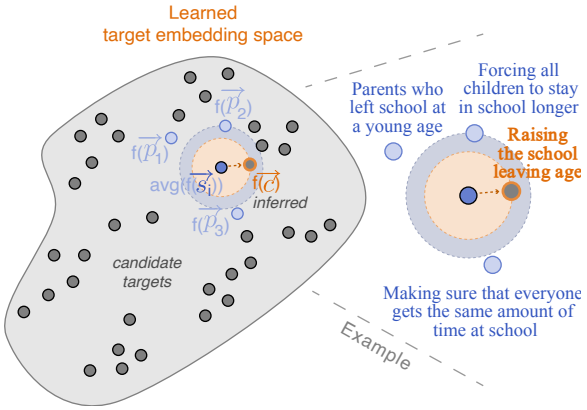

FIGURE 5.6: Sketch of inferring a conclusion target from an argument's premises. Given a knowledge base of candidates, the target is chosen whose learned embedding $f(\overrightarrow{c})$ is closest to the learned average $avg(f(\overrightarrow{s_i}))$ of premise targets. An example is shown on the right. Illustration is taken from Alshomary et al. (2020b).

A Hybrid of Both Sub-Approaches The reasonableness of the conclusion target inferred by the second sub-approach depends on the quality of the knowledge base. To avoid inferring fully unrelated targets, we also consider a simple hybrid of our two approaches below; if the target inferred by the embedding learning approach overlaps with the (full) text of any premise in at least one content token, it is taken. Otherwise, the target inferred by the premise ranking is taken.

Transfer Learning for Conclusion Generation

The advances in transformer-based language models lead to new state-of-the-art results on various NLP tasks. While the presented approach to inferring conclusions is suitable when training data is limited to training transformer-based language models, utilizing these models might be more suitable for other scenarios. Therefore, we investigate transformer-based models' effectiveness on the conclusion inference task. In particular, we fine-tune the BART model Lewis et al. (2020) on a corresponding dataset containing pairs of arguments and their conclusions in a single task setting. We additionally investigate the performance when the model is trained in a multi-task setting to predict a counter for the argument and its conclusion. Details on this part are provided in Subsection 5.2.2.

5.1.2 Weak Premise Identification

Since one popular method of countering the opponent's argument relies on attacking one of its weak premises (argument undermining), we argue that studying the task of identifying weak premises is relevant to the overall effectiveness of debate technology. The following section describes our work on identifying weak

premises in the opponent's argument. We define the task as follow. Given an argument in the form of a conclusion and a set of premises, the task is to identify the argument's attackable premises. Unlike previous work (Jo et al., 2020), we model the task as a ranking task instead of a classification task, in which, for each argument, we learn to rank its premises by their weakness relevant to the claim. Our hypothesis here is that the attackability of a premise can be better learned when considering both the claim and other premises of the argument. We operationalize the weak-premise ranking similar to the ranking approach of Han et al. (2020). In particular, given a set of premises and the conclusion, we represent each premise by concatenating its tokens with the conclusion's tokens, separated by special tokens *[cls]* and *[sep]*, as follows:

[cls] claim_tokens [sep] premise_tokens [sep]

Next, the resulting sequences are passed through a BERT model to obtain a vector representation for every premise. Each vector is then projected through a dense layer to get a score $\hat{y}$ that reflects the weakness of the premise. Finally, a list-wise objective function (we use a Softmax loss) is optimized jointly on all premises of an argument as follows:

$$L(y, \hat{y}) \;=\; -\sum_{i=1}^{n} y_i \cdot \log\Big(\frac{exp(\hat{y_i})}{\sum_{j=1}^{n} exp(\hat{y_j})}\Big),$$

where y is a binary ground-truth label reflecting whether the given premise is attackable ($y = 1$) or not ($y = 0$). Given training data, we can thus learn to rank premises by their weakness.

5.2 Counter Generation

In the following, we present two realizations of our approach's second component to integrate the knowledge learned from the previous section into the process of counter-argument generation. The first realization joins the argument analysis component with counter generation as a pipeline, where the identified weak premises from the first component are encoded into a transformer-based language model to generate counters that attack these weak premises (argument undermining). The second realization, however, jointly learns the two tasks of argument analysis and counter generation, where the conclusion is inferred as part of the argument analysis component, and the counter is generated in the counter generation component (argument rebuttal).

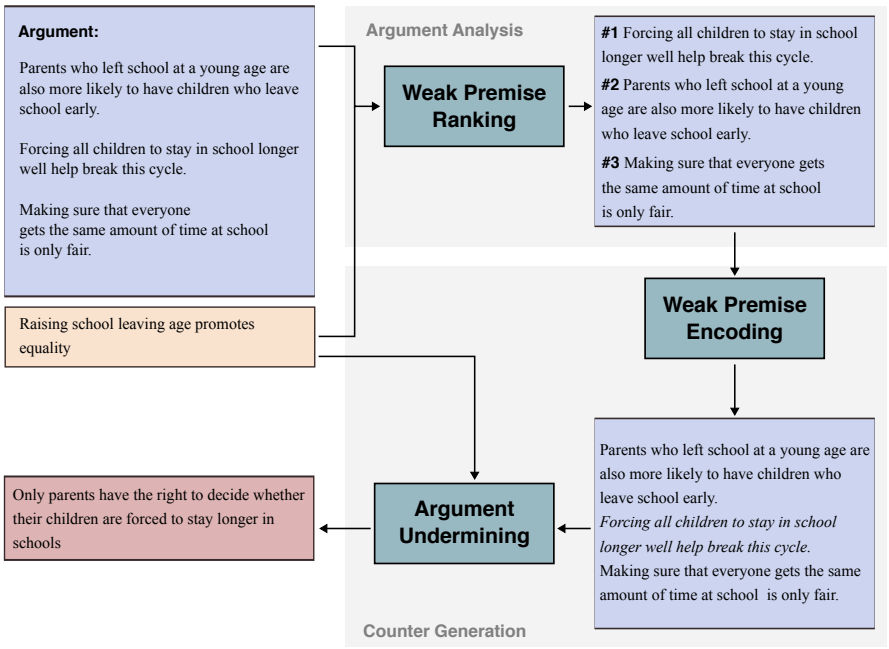

FIGURE 5.7: Our argument undermining approach, where we realize the two main steps of our approach (argument analysis and counter generation) as a pipeline. Given the input argument with its premises ranked according to their attackability, we encode this information on the token level (highlighted in italics), and the argument undermining component learns to counter the argument by attacking the highlighted premise.

5.2.1 Argument Undermining

As highlighted in Figure 5.7, given the output of the weak premise identification model as a ranking of premises, we identify the k highest-ranked ones to be attackable (in our experiments, we test $k = 1$ and $k = 3$). Then, we generate a counter-argument putting the identified attackable premises into the focus. To this end, we follow Wolf et al. (2019) in using transfer learning to fine-tune a pre-trained transformer-based generation model on our task. In our fine-tuning process, the input is a sequence of tokens created from two segments, the argument, and the counter-argument:

[bos] arg_tokens [counter] counter_tokens [eos]

The final token embedding is then a result of concatenating three embeddings: word and positional embeddings learned in the pre-training process, as well as a token-type embedding learned in the fine-tuning process. Here, the token type reflects whether the token belongs to the argument in general, to a weak premise, or to the counter-argument. Now, we train our model jointly on two tasks:

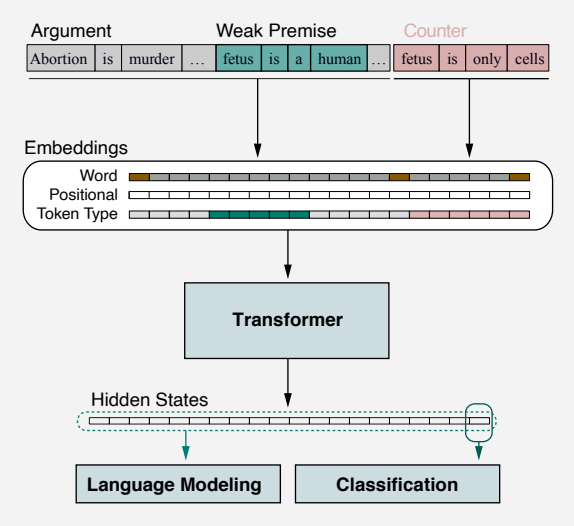

FIGURE 5.8: Architecture of our approach from Alshomary et al. (2021c): Given an argument, a weak premise, and a counter, three embedding representations are generated and fed to the transformer to obtain hidden states from which the language model and classification heads learn the *Next-token prediction* and *Counter-argument classification* tasks respectively..

- *Next-token prediction.* Given a sequence of tokens, predict the next one.
- *Counter-argument classification.* Given two concatenated segments, decide whether the second is a counter-argument to the first.

The first task is similar to the next-sentence prediction task introduced in Devlin et al. (2018), which was shown to be beneficial for representation-learning tasks.

Figure 5.8 shows the architecture of our generation model. For training, we augment a given set of training sequences D by adding distracting sequences. Concretely, we use, for each argument and its weak premise, a non-relevant text instead of the counter-argument. Given a sequence of tokens $d = (t_1, t_2, \cdots, t_n) \in D$, we then optimize the following two loss functions jointly with equal weighting:

$$L_1(\Theta) = \sum_{d \in D} \sum_{t_i \in d} \log P(t_i \mid t_{i-k}, \cdots, t_{i-1}; \Theta),$$

$$L_2(\Theta) = \sum_{d_j \in D} \log P(y_j \mid t_1, \cdots, t_n; \Theta),$$

where Θ denotes the weights of the model, k is the number of previous tokens, and y_j is the ground-truth label of the sequence, indicating if the second segment of the sequence is a counter or not.

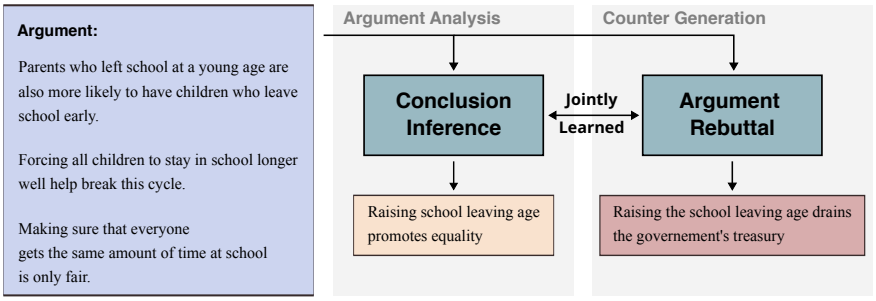

FIGURE 5.9: Our argument rebuttal approach, where the two main steps of our approach (argument analysis and counter generation) are learned jointly. Given the input argument, the conclusion inference and argument rebuttal models learn to generate the conclusion and the counter in a multitask fashion.

5.2.2 Argument Rebuttal

As mentioned above, conclusions are essential to understanding the reasoning behind arguments, enabling the generation of relevant counters. However, they often remain implicit, making understanding hard for machines. While we discussed extensively in Section 5.1.1 our approach to generate conclusions, the advances in language models brought new state-of-the-art results to the task. Therefore, in our argument rebuttal approach, we focus on utilizing these models for the task. Particularly, as highlighted in Figure 5.9, in Alshomary and Wachsmuth (2023), we proposed an approach that jointly learns to generate an argument's conclusion and counter. At inference time, it employs a stance-based ranking component that selects the candidate counter that best counters the synthesized conclusion. We will detail our approach's generation and ranking steps in the following.

Joint Generation

Text generation is usually modeled as a sequence-to-sequence task and is often addressed through transformer-based encoder-decoder models (Vaswani et al., 2017). As illustrated in Figure 5.10, we realize our approach of jointly learning the two tasks in two ways - sharing the full model between the two tasks or only the encoder part. We will explain each option in the following.

Fully-shared Encoder and Decoder In the first model, we maintain the same transformer-based encoder-decoder architecture and train it to generate output sequences containing both the conclusion and the counter. Hence, the model learns to perform the two tasks simultaneously. Particularly, the input to the model is one sequence representing an argument's premises, and the output is a single sequence composed of the ground-truth conclusion and counter-argument separated by spe-

cial tokens, <conclusion> and <counter>. The model encodes premises and decodes first the conclusion and then the counter in one sequence. We train the model to optimize the following loss function:

$$L(\theta) = -\sum_{1}^{n} \log p(y_i|x, y_{<i}; \theta)$$

Here, x is the input sequence that represents the premises, $y_{<i}$ is the sequence composing the conclusion and counter until the next word y_i, and θ denotes the model's parameters. We call this model *Joint One-seq* later in our experiments.

At inference time, we utilize a mechanism to generate a diverse set of n candidate conclusions and their counter-arguments, which are later passed to our stance-based ranking component to select the best counter. The diverse generation is as follows. We first extract a set of m Wikipedia concepts from the input premises using the approach of Dor et al. (2018). Then, during decoding, we use these concepts to prompt our trained model by masking all logits except the ones matching the prompt tokens, resulting in conclusions addressing different aspects of the premises followed by their corresponding counters. Moreover, to ensure candidate diversity, we enable nucleus sampling (Holtzman et al., 2019), where at each step, we randomly select one of the top k tokens with an accumulated probability of more than p.

Shared Encoder with two Decoders Similarly, the second model starts with an argument's premises as input. However, it then decodes two independent sequences representing the conclusion and the counter-argument as output. First, the input premises are passed through a shared encoder, and then two decoders are used to learn to generate the counter and the conclusion. During training, we optimize the following multi-task loss function, which is a weighted average of the two language modeling losses of the two decoders:

$$\begin{aligned} L(\theta_e, \theta_a, \theta_b) = \quad & \alpha_a \cdot \sum_{i=1}^{n} \log p(y_i^{(a)}|x, y_{<i}^{(a)}; \theta_e; \theta_a) \\ + \ & \alpha_b \cdot \sum_{i=1}^{m} \log p(y_i^{(b)}|x, y_{<i}^{(b)}; \theta_e; \theta_b) \end{aligned}$$

Here, $y^{(a)}$ and $y^{(b)}$ are the conclusion and counter sequences. θ_e, θ_a, and θ_b are the weight parameters of the encoder, the conclusion decoder, and the counter decoder, respectively. The weights, α_a and α_b, sum up to one. Their best values are determined experimentally during validation.

The difference between this model and the previous one is given by the layers shared between the two tasks. In the previous model, both the encoder and decoder

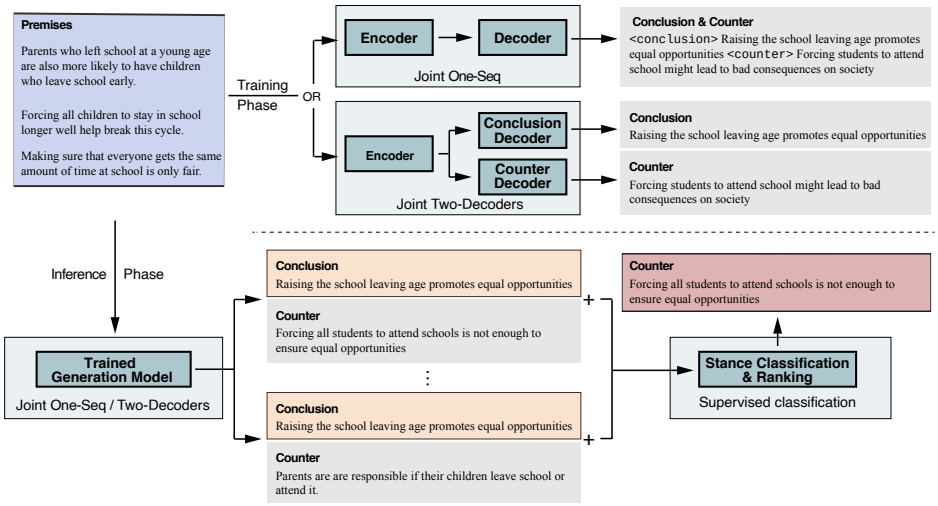

FIGURE 5.10: Both variations of our proposed approach to argument rebuttal. In the *training phase*, we learn to jointly generate the conclusion and counter either as one sequence (*Joint-based One-seq*, variation 1) or as two separated sequences (*Joint-based Two-decoders*, variation 2). In the *inference phase*, we classify and rank a diverse set of counters with respect to their stance towards the corresponding conclusion. The top-ranked counter is used (Alshomary and Wachsmuth, 2023)

layers are shared between the two tasks, while here, only the encoder's layers are shared, keeping a dedicated decoder for each of the two tasks. We refer to this model as *Joint Two-decoders* below.

We aim to generate a diverse set of candidate counters similar to the above model. However, we noticed that counters rarely start by referring to entities or similar concepts, and prompting the model with concepts might lead to generating irrelevant texts. Hence, we generate one conclusion for this model but a set of candidate counters by only enabling the nucleus sampling during decoding.[1]

Ranking Component

Given a set of n generated candidate counters, we rank them based on their stance contrastiveness towards the corresponding generated conclusion and select the top-ranked as our final output. In particular, we trained a transformer-based stance classifier on pairs of claim and counter-claim acquired from the *kialo.com* platform to be used to predict whether the pair has a *pro* or *con* stance. Experimental details are provided in the next Section. To guarantee the stance coherence of the selected counter, we compute the stance-based scores on the sentence level to ensure all sentences have some degree of contrastiveness towards the conclusion. In particular, given a pair of a conclusion and the corresponding counter, we first split the

[1] We tested the performance of the model empirically and noticed that these prompted counters of low quality.

counter into a set of sentences. For each sentence s_i, we apply our trained classifier to compute the stance $label$ towards the conclusion c and its probability pr_{label}. We then translate this into a stance contrastiveness score as follows:

$$cont(s_i, c) = \begin{cases} pr_{con}, & \text{if } label = con \\ -pr_{pro}, & \text{if } label = pro \end{cases}$$

The final score of a counter is averaged across its sentences, ranging from -1 to 1. The counters are then ranked accordingly, selecting the top one.

5.3 Evaluation

In the following, we will present a series of experiments performed to evaluate each part of our approach. First, we will investigate our hypothesis regarding the relation between premises and conclusion targets and the effectiveness of our methods. We will then evaluate our weak premise ranking method compared to the related work of Jo et al. (2020). Next, we will assess the effectiveness of our counter-argument generation approaches. Particularly, we will investigate whether integrating the knowledge about the weak premises and conclusions of arguments leads to generating more effective counters.

5.3.1 Conclusion Inference

In the following, we give details on a series of experiments we carried on in our previous work by Alshomary et al. (2020b) to evaluate our hypothesis on the importance of identifying premise targets in the process of conclusion generation.

Data

In our evaluation, we use three datasets; one to train the target identification model (*Claim Stance Dataset*), and two to evaluate our approaches for the main task of target inference (*iDebate Dataset* and *AAE*). In particular, the *Claim Stance Dataset* (Bar-Haim et al., 2017) contains 2,394 claims referring to 55 topics from Wikipedia articles. Not only the stance of premises towards their topics is manually annotated, but also a phrase is marked in each claim as being a target. We use this dataset to train and evaluate a target phrase tagging model for *identifying* targets in the given premises of an argument. As Bar-Haim et al. (2017), we take all premises associated with 25 conclusions for training and the rest for testing. The *iDebate Dataset* (Wang and Ling, 2016) consists of 2,259 pro and con points for 676 controversial issues from the online debate portal *idebate.org*. Each point comes with a one-sentence conclusion (called *central claim* by the authors) and an argumentative text supporting the conclusion. Each sentence is seen as one premise of the conclusion (called *argument*), resulting in a total of 17,359 premises. We use this dataset for

training, optimizing, and evaluating all approaches to conclusion target inference. Following its authors, we split the dataset based on debates: 450 debates for training, 67 for validation, and 150 for testing. Finally, the *Argument Annotated Essays (AAE)* corpus (Version 2; Stab and Gurevych (2014b)) includes 402 persuasive student essays. Each essay was segmented manually into subsentence-level argument components: theses (called *major claims*), conclusions (*claims*), and premises. We use this corpus to study target inference in a second domain. To analyze different types of argument relations, we derive two datasets from the corpus: *Essay Conclusions* for conclusions and their premises with 1,530 training, 256 validation, and 234 test cases, and *Essay Theses* for theses and the underlying conclusions with 300 training, 50 validation, and 52 test cases.

Automatic Evaluation

Premise Target Identification We implemented the target identifier as a BiLSTM-CRF with hidden layer size 256, using the pre-trained contextual string embedding model of Akbik et al. (2018). We trained the model on the training set of the Claim Stance Dataset with batch size 16 and a learning rate of 0.1 for five epochs. The identifier achieved an F1-score of 0.77 on the Claim Stance test set. To assess its effectiveness in other domains, we let human annotators evaluate the identified targets of a random sample of 100 conclusions from the iDebate dataset. Three annotators evaluated each instance. Based on the majority agreement, the tagger identified 72% of the cases correctly. In terms of Fleiss' κ, the agreement was 0.39, which is low but reasonable, given that we did not train annotators.

Conclusion Target Inference To evaluate target inference, we use the iDebate Dataset and the two essay datasets. As no ground-truth conclusion targets are provided, we used our target identifier to extract targets from the conclusions and compared them to the output of our approaches. In some cases, particularly where targets were not explicitly phrased, our target identifier did not annotate any token. Hence, we eliminated those cases from the test set.[2]

Implementation Details For the premise target ranking approach, we trained LambdaMART (Burges, 2010) on each training set with 1000 estimators and a learning rate of 0.02. We refer to this approach below as *Premise Targets (ranking)*. For target embedding learning, we used the pre-trained FastText embeddings with 300 dimensions (Bojanowski et al., 2017) to initially represent each target. To obtain a knowledge base of candidate targets, we applied the target identifier to all conclusions of all training sets. The resulting lexicon contains 1,780 targets. Each

[2]Example conclusion where no target was identified: *"It makes it more difficult for extremists to organize and spread their message when blocked"*.

#	Approach	Scenario	iDebate dataset		
			bleu	**meteor**	**accur.**
b1	Seq2Seq	–	0.7	0.01	0%
b2	Seq2Seq (w/ premise targets)	–	4.4	0.07	5%
b3	Premise Targets (random)	–	3.9	0.11	8%
b4	Target Embedding (average)	Optimistic	7.2	0.16	18%
		Pessimistic	6.4	0.15	17%
a1	Premise Targets (ranking)	–	9.7	**0.16**	17%
a2	Target Embedding (learning)	Optimistic	9.2	0.15	18%
a2		Pessimistic	7.2	0.13	16%
a1&a2	Hybrid	Optimistic	**10.0***	**0.16**	**20%***
		Pessimistic	8.1	0.15	18%
	Oracle (upper bound)	Optimistic	94.3	0.85	100%
		Pessimistic	35.8	0.58	65%

TABLE 5.1: Effectiveness of the evaluated target inference approaches in terms of BLEU, METEOR, and accuracy on the test sets of the iDebate dataset. The best value in each column is marked bold. Values of *a1&a2* marked with * are significantly better than the best baseline *b4* at $p < 0.05$ (student t-test). The bottom rows show the effectiveness of an oracle that selects those conclusion targets, which maximize each score.

target is represented by its FastText embedding. We implemented the triplet neural network as three feed-forward neural networks, each with two layers and shared weights. We call this approach *Target Embedding (learning)*. The simple hybrid of both approaches introduced above is denoted *Hybrid (ranking & embedding)*. As for the baselines, on the one hand, we compare them to the state-of-the-art sequence-to-sequence argument summarizer, at that time (Wang and Ling, 2016). Since its code is not available, we approximately reimplemented it. Specifically, we replicated the importance sampling with the same features (also on five premises) but no regularization. We used three LSTM layers with the hidden size 150 and a pre-trained embedding of size 300 to perform text generation. Extra features of the original approach were left out, as they did not help much in our case. We trained the model with batch size 48 and a learning rate of 0.1 using the Adagrad optimizer. To identify targets in the generated summaries, we employed our target identifier. We refer to this baseline as *Seq2Seq*. To test our hypothesis on the relation of premise and conclusion targets, we extended *Seq2Seq* by a pointer generator (See et al., 2017) and an extra binary feature that encodes whether a token belongs to a target or not, allowing the model to learn this relation. We call this *Seq2Seq (w/ premise targets)*. On the other hand, we complemented our approaches with simpler variants in order to check whether learning is needed. Instead of premise target ranking, our baseline *Premise Targets (random)* simply chooses a premise target randomly. Instead of target embedding learning, we simply pick the target

#	Approach	Scenario	Essay Conclusions			Essay Theses		
			bleu	meteor	accur.	bleu	meteor	accur.
b3	Premise Targets (random)	–	2.2	0.09	3%	8.8	0.19	17%
b4	Target Embedding (average)	Optimistic	**8.3**	0.12	**8%**	15.3	0.24	21%
		Pessimistic	4.1	0.12	6%	15.3	0.24	21%
a1	Premise Targets (ranking)	–	4.1	0.11	5%	17.3	0.25	24%
a2	Target Embedding (learning)	Optimistic	**8.3**	0.12	**8%**	**27.9**	**0.29**	**27%**
a2		Pessimistic	3.4	0.09	5%	13.6	0.23	21%
a1&a2	Hybrid	Optimistic	8.2	**0.13**	**8%**	**27.9**	**0.29**	**27%**
		Pessimistic	3.4	0.10	5%	13.6	0.23	21%
	Oracle (upper bound)	–	98.9	0.95	100%	98	0.90	100%
		Pessimistic	34.2	0.59	49%	26	0.52	48%

TABLE 5.2: Effectiveness of the evaluated target inference approaches in terms of BLEU, METEOR, and accuracy on the test split of the two essay datasets. The best value in each column is marked bold. Values of *a1&a2* marked with * are significantly better than the best baseline *b4* at $p < 0.05$ (student t-test). The bottom rows show the effectiveness of an oracle that selects those conclusion targets, which maximize each score.

from the target space whose embedding is most similar to the average premise target embedding, called *Target Embedding (average)*.

Experimental Setup We tuned all approaches on the respective validation sets, and then evaluated them on the test set. Since *Seq2Seq* requires much training data, we evaluated both variants on iDebate only. Before the inference of *Target Embedding (learning)*, the corresponding premise targets were added to the knowledge base as candidates for a conclusion target. Below, we consider two scenarios, an *optimistic* and a *pessimistic* one. The ground-truth target is added to the knowledge base in the former but not in the latter. The optimistic scenario thus reflects the effectiveness of the approach regardless of the limitations of the knowledge base. As for the measures, we use two common complementary evaluation measures, BLEU and METEOR. BLEU counts n-gram matches (we include 1- and 2-grams) focusing on precision, while METEOR is recall-oriented. Additionally, we also report accuracy, where a given target is correct if it has 50%+ content overlap with the ground truth.

Results Table 5.1 and 5.2 list our results. Clearly, encoding premise targets into *Seq2Seq* boosts its effectiveness, indicating the importance of modeling premise targets (Table 5.1). However, both *Seq2Seq* variants perform poorly compared to our approaches. While the limited training data size is one reason, this also indicates that pure sequence-to-sequence generation may not be enough.

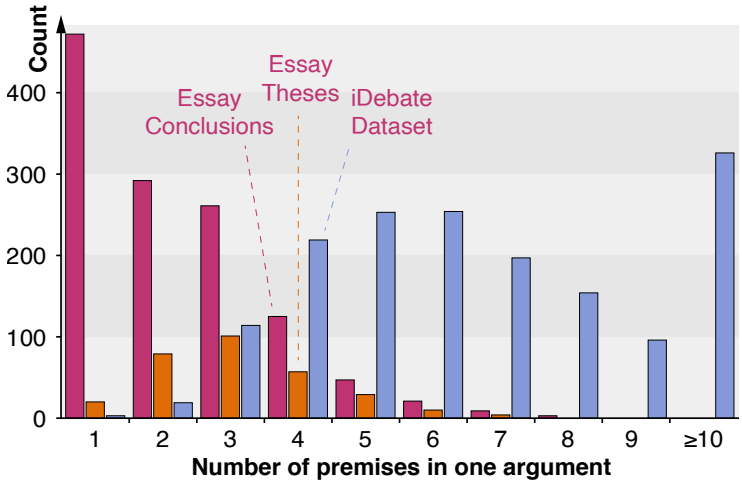

FIGURE 5.11: Histogram of the number of arguments with a specific number of premises in the three given datasets. Illustration is taken from Alshomary et al. (2020b).

#	iDebate dataset		Essay Conclusions		Essay Theses	
	New	**Exact**	**New**	**Exact**	**New**	**Exact**
b4	5%	6%	3%	2%	0%	6%
a1	0%	**9%**	0%	1%	0%	9%
a2	**24%**	7%	**25%**	**6%**	**12%**	**12%**
a1&a2	9%	8%	15%	**6%**	**12%**	**12%**

TABLE 5.3: Percentage of test cases where each approach picked a *new* target (not a premise target) and where the picked target is an *exact* match of the ground-truth target. The highest value in each column is marked bold.

As shown in Table 5.1, on iDebate, both approaches are better than all baselines in terms of BLEU score. The best results are achieved by *Hybrid* in terms of all measures (significantly for BLEU and accuracy). Even in the pessimistic scenario, its BLEU score of 8.1 outperforms all baselines. In the optimistic scenario on the essay datasets (Table 5.2)), *Target Embedding (learning)* is strongest for most scores. The hybrid approach hardly achieves any improvement. Due to the small dataset size, no significance was found, though. In the pessimistic scenario, *Premise Target (ranking)* seems more suitable. The lower scores on Essay Conclusions can be attributed to the low number of premises (see Figure 5.11), which makes finding an adequate conclusion target among the premise targets less likely. However, all approaches are much worse than theoretically possible (*oracle*) in terms of automatic metrics.

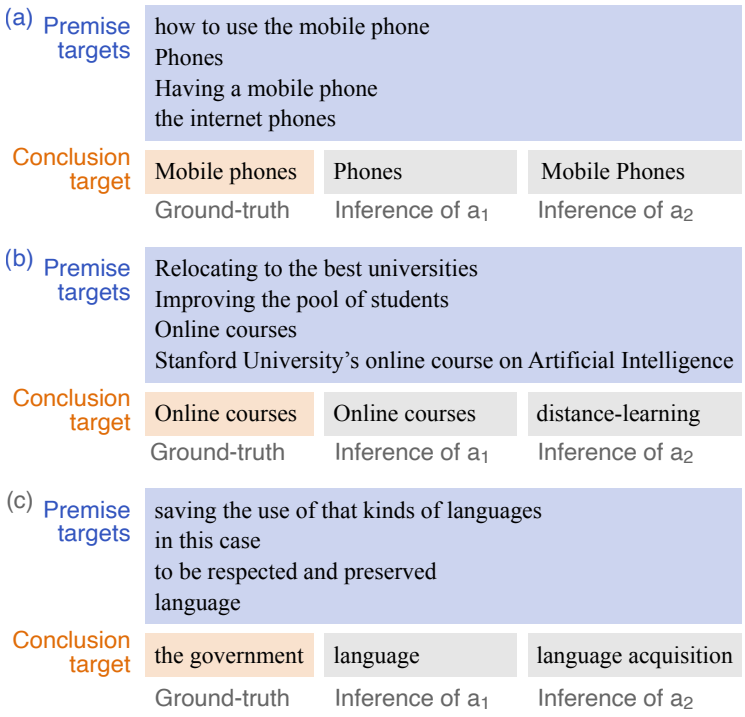

FIGURE 5.12: Three examples of premise targets from the datasets, the associated ground-truth conclusion target, and the conclusion targets inferred by our approaches. Illustration is taken from Alshomary et al. (2020b).

Analysis To illustrate the behavior of selected approaches, Table 5.3 compares the percentages of cases where they pick a new target as well as where they pick the exact ground-truth conclusion target (in the optimistic scenario). Befittingly, target embedding learning (*a2*) is most "exploratory" regarding new targets. On the essay datasets, where the conclusion target only sometimes occurs in the premises, *a2* is also best in inferring the exact target. Still, premise target ranking (*a1*) may pick the ground truth if it matches any premise target. The hybrid seems a suitable balance between both.

Figure 5.12(a) exemplifies the ability of *a2* to infer the correct conclusion target even if it does not match a premise target exactly. Example (b) stresses the limitation of automatic evaluation: "distance-learning" (inferred by *a2*) does not overlap with the ground truth, but it semantically matches well. In (c), the ground-truth target was barely inferable from the premise targets. Full example arguments can be found in the Appendix (Table A.7).

#	Scenario	Fully	Somewhat	Not	Majority
b2	–	5%	18%	76%	93 / 100
a1	–	56%	33%	11%	91 / 100
a2	Optimistic	50%	28%	22%	92 / 100
	Pessimistic	49%	27%	24%	93 / 100
a1&a2	Optimistic	55%	34%	11%	89 / 100
	Pessimistic	56%	32%	12%	90 / 100
Ground-truth		62%	29%	10%	84 / 100

TABLE 5.4: Majority agreement for how adequate (*fully*, *somewhat*, *not*) are the conclusion targets of baseline *b2*, our approaches, and the ground truth. The right column lists the number of cases where majority is given.

Manual Evaluation

To assess the actual quality of the inferred conclusion targets, we manually evaluated our approaches (optimistic and pessimistic scenario) and the baseline *b2* (*Seq2Seq (w/ premise targets)*) in comparison to the ground-truth targets using Amazon Mechanical Turk. We sampled 100 random instances from the iDebate test set. In a single task, an argument's premises were given along with the conclusion target of either approach. Annotators had to judge the adequacy of the target for the given premises as *fully*, *somewhat*, or *not* adequate. Each instance was judged by five annotators. No one judged multiple targets for the same argument.[3]

Table 5.4 shows the distribution of majority judgments for each approach. Only 23% of the *b2* targets were considered fully or somewhat adequate, i.e., pure text generation seems insufficient. In contrast, our sub-approaches' targets are competitive to the ground truth, which was not always adequate either (likely due to errors in target identification). The high performance of *a1* (*Premise Targets (ranked)*) might be explained by the inferred targets being part of the premises, affecting annotators' preferences. Still, the targets of *a2* (*Target Embedding (learning)*) are seen as adequate in 78% of the cases (50% fully), with the ability to infer conclusion targets that are not explicitly stated in the premises. Even in the pessimistic scenario, the inferences of *a1* and *a1&a2* remain stable. These results indicates the importance of modeling the relation between premise and conclusion targets in order to generate adequate conclusions.

5.3.2 Identifying Weak Premises

In the following, we will discuss the evaluation of our approach to identifying weak premises, including the dataset we use for evaluation, the implementation details, and the results.

[3] We paid $0.40 per task, restricting access to annotators with an approval rate of at least 95% and 5000 approved tasks. We ensure correct annotations by demanding an explanation for each judgment.

Data

As proposed, our argument-undermining realization models the task of counter-argument generation as an attack on a potentially weak premise. Such behavior is widely observed on the Reddit forum *changemyview (CMV)*. In particular, a user writes a new *post* that presents reasons supporting the pro or con stance towards a given topic (captured in the *title* of the post), asking the CMV community to challenge the presented view. In turn, other users quote specific segments of the post (usually a few sentences) and seek to counter them in their *comments*.

The structure induced by CMV defines a suitable data source for our study the task of identifying weak premises and later the argument undermining task. Specifically, we create the following distantly-supervised mapping:

- The title of the post denotes the *claim* of the user's argument;
- the text of the post denotes the concatenated set of the argument's *premises*;
- the quoted sentence(s) denote the *attackable (weak) premises*; and
- the quoting sentences from the comment denote the *counter-argument*.

In our work, we build on the CMV dataset of Jo et al. (2020), where each instance contains a post, a title, and a set of attackable sentences (those quoted in the comments). We use the same split as the authors, consisting of 25.8k posts for training, 8.7k for validation, and 8.5k for testing. To use this dataset for the task of a counter-argument generation later, we extend it by further collecting the quoting sentences from the comments (i.e., the counter-arguments). The final dataset compiles 111.9k triples of argument (claim and premises), weak premise (one sentence or more), and counter-argument (a set of sentences), split into 67.6k training, 23k validation, and 22.3k test instances.

Implementation Details

As presented, we tackled the task of finding attackable premises by learning to rank premises by their weakness with respect to the main claim. Based on the code of Han et al. (2020) available in the Tensorflow learn-to-rank framework (Pasumarthi et al., 2019), we used a list-wise optimization technique that considers the order of all premises in the same argument. Additionally, we also experimented with point-wise and pairwise techniques, but the list-wise approach turned out best. We trained our ranking approach on the CMV dataset's training split, and we refer to it as *bert-ltr* below. We compare our approach to the Bert-based classifier introduced by Jo et al. (2020), trained on the same training split using the authors' code. We employed their trained model to score each premise and then rank all premises in an argument accordingly. We call this the *bert-classifier*. As Jo et al. (2020), we also consider a *random baseline* as well as a baseline that ranks premises based on *sentence length*.

Approach	P@1	A@3
Random	0.425	0.738
Sentence Length	0.350	0.617
bert-classifier (Jo et al., 2020)	0.487	0.777
bert-ltr (our approach)	***0.506**	***0.786**

TABLE 5.5: Weak-premise ranking: Precision of ranking a weak premise highest (P@1) and accuracy for the top three (A@3) of all evaluated approaches. Results with * are significantly better than *bert-classifier* at $p < .05$.

Results

To assess the effectiveness, we follow Jo et al. (2020) in computing the precision of putting a weak premise in the first rank ($P@1$), as well as the accuracy of having at least a weak premise ranked in the top three ($A@3$). Table 5.5 shows the weak-premise ranking results. We managed to almost exactly reproduce the values of Jo et al. (2020) for all three baselines. Our approach, *bert-ltr*, achieves the best scores according to both measures. In terms of a one-tailed dependent student's t-test, the differences between *bert-ltr* and *bert-classifier* are significant with at least 95% confidence. These results support our hypothesis of the importance of tackling the task as a ranking task with respect to the main claim. Later, we then use our weak-premise ranking model in the argument-undermining approach to generate counter attacking the identified weak premises.

5.3.3 Argument Undermining

Next, we evaluate our hypothesis on the importance of identifying weak premises in counter-argument generation. In Alshomary et al. (2021c), we conduct two experiments. First, we use the ground-truth weak premises from our data and focus on evaluating the encoding method. In the second experiment, we evaluate the overall argument-undermining approach to predict the weak premises and then encode them into the language model to generate the final counter. In the following, we present the implementation details for the two experiments.

Argument Undermining with Ground Truth Weak Premises

Implementation Details We used OpenAI's GPT as a pre-trained language model. We trained two versions of our generation model: *our-model-w/* with an extra special token (*[weak]*) surrounding the attackable sentences to give an extra signal to our model, and once *our-model-w/o* without it. We fine-tuned both versions with the same settings using the transformers library (Wolf et al., 2020) for six epochs.[4] We left all other hyperparameters with their default values. As men-

[4]We stopped at six epochs because we observed no gain in terms of validation loss anymore.

		Counter Sentences			Full Comment		
Approach	**Target**	**MET.**	**BLEU-1**	**BLEU-2**	**MET.**	**BLEU-1**	**BLEU-2**
counter-baseline	-	0.058	13.023	3.117	0.097	10.400	3.212
our-model-w/o	claim	**0.060**	12.532	2.943	0.090	9.472	2.837
our-model-w/o	random premise	0.058	12.838	3.005	0.096	10.398	3.255
our-model-w/o	weak premise	0.057	*__13.453__	*__3.391__	*__0.102__	*__10.998__	*__3.764__
our-model-w/	claim	**0.060**	12.635	3.023	0.092	9.685	2.984
our-model-w/	random premise	0.059	12.712	2.987	0.096	10.161	3.217
our-model-w/	weak premise	0.058	13.162	3.217	0.101	10.743	3.651

TABLE 5.6: Premise attack generation: METEOR (MET.), BLEU-1, and BLEU-2 scores of the output of each evaluated approach compared to the ground-truth *counter sentences* and to the *the full comment* (i.e., the full counter-argument). Values marked with * are significantly better than *counter-baseline* at $p < .05$.

tioned, the model's input is a sequence of tokens constructed from the argument (with weak premises highlighted) and either the correct counter or a distracting sequence. We selected one sentence from the original post randomly to be the distracting sequence for each input instance. We compare our model to a GPT-based model fine-tuned on a sequence of tokens representing a pair of an argument (title and post) and a counter-argument. We consider this as a general counter-argument generation model, trained without any consideration of weak premises. We trained the baseline using the same setting as our model. We refer to it as *counter-baseline*.

Automatic Evaluation To assess the importance of selecting attackable sentences, we evaluate the effectiveness of our model in different inference settings in terms of what is being attacked: (1) the *claim* of the argument, (2) a *random premise*, or (3) a *weak premise* given in the ground-truth data. For the random setting, we selected three premises from the argument randomly, and we generated one counter for each. The final result is the average of the results for each. We computed METEOR and BLEU scores, comparing the generated premises to (a) the exact counter sentences of the quoted weak premise and (b) the full argument. We carried out this automatic evaluation on 1k posts from the test split.

As shown in Table 5.6, the best results are achieved by *our-model-w/o* in all cases when identifying the weak premises in the input. Encoding the knowledge about weak premises as token types is sufficient, and adding an extra special token does not help. Although the differences between our best model and the baseline are not big, they are significant according to the one-tailed dependent t-test with a confidence of 95%. For both versions of our model, the best scores are achieved when considering the weak premises as the target (except for the first METEOR column). However, not all these differences are significant. This gives evi-

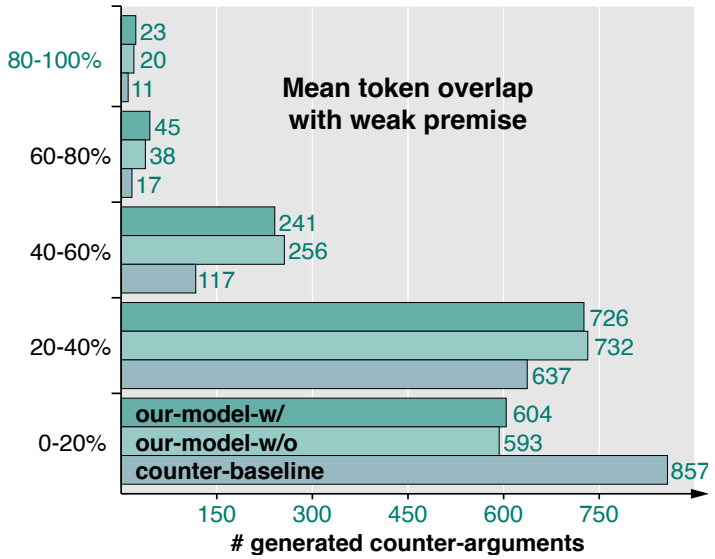

FIGURE 5.13: Premise attack generation: Mean token overlap between the ground-truth weak premises and the counters generated by each evaluated approach. Illustration is taken from Alshomary et al. (2021c).

dence that exploiting information about weak premises in the training of counter-argument generation approaches can improve their effectiveness.

To further assess the relationship between the generated counters and the attacked premises, we computed the proportion of covered content tokens in the weak premise for the two versions of our model and the baseline. Figure 5.13 shows a histogram of the percentages. Clearly, both versions of our model have higher coverage of the annotated weak premises than the baseline.

Manual Evaluation To analyze the generated counter-arguments more thoroughly, we carried out a manual evaluation study on a sample of 50 random examples. Two authors of our work (Alshomary et al., 2021c) inspected the sample, comparing the two versions of our model. The results were in favor of *our-model-w/o*. Therefore, we compared only *our-model-w/o* against the *counter-baseline*. In particular, we assessed the relevance and appropriateness of the output of the two for each example. Given an argument, the highlighted premise to be attacked, and the two counters, we asked three annotators who hold an academic degree and are fluent in English (no author of this paper) to answer two questions:

1. Which text is more relevant to the highlighted premise?
2. Which text is more appropriate for being used as a counter-argument?

As shown in Table 5.7, annotators favored our model in 56%

Approach	Relevance		Appropriateness	
	Majority	**Full**	**Majority**	**Full**
counter-baseline	44%	20%	44%	14%
our-model-w/o	**56%**	**36%**	**56%**	**28%**
Mean Kendall's τ	0.41		0.23	

TABLE 5.7: Premise attack generation: Percentage of cases where each given approach was seen as more relevant and more appropriate, respectively, according to majority vote and the full agreement in the manual evaluation on 50 examples. The bottom line shows the mean pairwise inter-annotator agreement.

Considering the given task as a ranking task, we used Kendall's τ to compute the annotator's agreement. The mean pairwise agreement was 0.41 for the relevance assessment and 0.23 for appropriateness. Clearly, assessing the text's appropriateness of being a counter-argument is more subjective and more challenging to judge than the relevance task.

Argument Undermining with Predicted Weak Premises

Finally, we assess the overall effectiveness of our proposed undermining approach to counter-argument generation; that is, we first identify weak premises automatically using our ranking model, *Bert-ltr*, and then generate a counter-argument using our generation model, *our-model-w/o*, focusing on the selected weak premises.

Implementation Details Due to the limited *P@1* value of our ranking model (see Table 5.5), we evaluate two variations of our overall approach that differ in terms of what premises to attack. The first variant attacks the weakest premise. In the second, we first generate three counters considering each of the top three weak premises. Then, we select the counter that has the most content-token overlap with the corresponding weak premise. On the one hand, we compare our approach to the *counter-baseline* from the previous section. On the other hand, we consider the state-of-the-art counter-argument generation approach of Hua and Wang (2019), an LSTM-based Seq2seq model with two decoders, one for selecting talking points (phrases) and the other for generating the counter, given the selection.

Automatic Evaluation While the approach of Hua and Wang (2019) learns from a dataset collected from the same source (CMV), it requires retrieving relevant argumentative texts with a stance opposite to the input argument. Due to the complexity of the data preparation, we decided instead to evaluate all approaches on the test split of Hua and Wang (2019).[5] As a result, the approach of Hua and Wang

[5]We verified that all posts in their test split do not appear in our training split.

#	Approach	Target	METEOR	BLEU-1	BLEU-2
1	counter-baseline	None	0.205	22.741	7.792
2	Hua and Wang	None	**0.258**	**30.160**	**13.366**
3	Overall approach	1 premise	0.207	22.841	7.839
4	Overall approach	3 premises	0.210	23.400	8.025

TABLE 5.8: Overall approach: METEOR and BLUE scores of the two variants with different attacked targets, the counter-baseline, and Hua and Wang (2019).

(2019) is trained on their training split, whereas our approach is trained on our training split, but both are then evaluated on the same test split of Hua and Wang (2019). This is a somewhat unfair setting for our approach due to certain domain differences, namely, the dataset of Hua and Wang (2019) comprises political topics only. Similar to Section 5.3.3, we generated counters for 1k examples and computed METEOR and BLEU scores of the generated counters with respect to the ground-truth counters, which are here full arguments (CMV comments).

Table 5.8 shows that our approach outperforms the counter-baseline in both settings, even with weak premises selected automatically. Considering the top-3 weak premises instead of the top-1 improves the results. The best scores are achieved by Hua and Wang (2019), though. A reason for this may be the slight domain difference between our model's training data and the test data used for evaluation. Another observation is that the scores of both our approach and the baseline increase compared to Table 5.6. This is likely to be caused by the higher number of ground-truth references for each instance in the dataset of Hua and Wang (2019) compared to the test split of our data, making it more likely to have the token overlaps.

Manual Evaluation Given the known limited reliability of automatic generation evaluation, we conducted another user study to evaluate the quality of the generated counters by our model and the approach of Hua and Wang (2019). We evaluate the same quality dimensions the authors used:

- *Content Richness.* The diversity of aspects covered by a counter-argument.
- *Correctness.* The relevance of a counter-argument to the given argument and their degree of disagreement.
- *Grammaticality.* The grammatical correctness and fluency of a counter-argument.

We used the Upwork crowd working platform to recruit three annotators with English proficiency and experience in editorial work.[6] We asked each of them to evaluate a sample of 100 examples. Each contained an argument (claim and premises)

[6]Upwork, `http://upwork.com`

	Correctness	Richness	Grammaticality
Hua and Wang	1.81	2.28	2.91
Overall approach	**2.65**	**3.15**	**3.50**
Krippendorff's α	0.26	0.06	0.32

TABLE 5.9: Overall approach: Average scores of the three annotators for the three evaluated quality dimensions of the counter-arguments generated by our approach and the one of Hua and Wang (2019). 1 is worst, 5 is best. The bottom line shows the inter-annotator agreeement.

and two counters (one of each approach). We asked the annotators to compare the counters and to assess each with a score from 1 (worst) to 5 (best) for each quality dimension.

The results are presented in Table 5.9. Unlike in the automatic evaluation, the annotators gave, on average, higher scores on all quality dimensions to our generated counters than to those of Hua and Wang (2019).[7] Bringing knowledge from pre-trained language models (GPT) generally seems to contribute to the *grammaticality* and the *richness* of the generated counters. In terms of generating a *correct* counter, focusing the generation model on a specific weak premise in an argument seems to help (2.65 vs. 1.81), even though the results are far from perfect. Manual inspection revealed that far from all generated arguments are counters to exactly what is in the argument, indicating more room to work on this topic.

The Krippendorff's α values show that the annotators had a fair agreement on *grammaticality* and *correctness* (given the subjectiveness of the tasks), but only slight agreement on *content richness*. We, therefore, think that the results for the latter should not be overinterpreted.

In Figure 5.14, we show an example argument in favor of *income inequality*. Our approach considers the premise "being poor does not entitle someone to the cash of the rich people". It then generates a counter-argument on the topic of inequality, focusing on the fact that "being poor limits the ability to contribute to society". In contrast, the counter-argument generated by Hua and Wang (2019) diverges to address "low-income housing", which is less relevant to the topic. More examples of generated counters can be found in Appendix (Table A.8).

5.3.4 Argument Rebuttal

This subsection will describe the experiments carried out in our previous work (Alshomary and Wachsmuth, 2023) to investigate the importance of conclusion inference for the task of counter-argument generation. First, we will present the

[7] We note that the scores of Hua and Wang in Table 5.9 are notably lower than those reported by Hua and Wang (2019). We believe this to be due to the comparison with our approach that affected the annotator's scores.

Claim: there's nothing wrong with income inequality.
Premises: billionaires like the rockefellers and trump worked hard to earn their money and provide their families with luxury. meanwhile, my grandpa didn't and because of that i am of lower-middle class status. *just because i'm poorer doesn't mean i'm entitled to the cash that the rich spent years to accumulate*. i simply have to swallow my pride and start at the bottom . to try and stump income inequality is to meddle with the very basis of pure , unadulterated capitalism and meritocracy
Our Counter: income inequality is precisely because of the fact that rich people have a vested interest in having a bunch of money in the first place, which leads to great wealth inequality. if you are poor but poor you have no incentive to live. by providing good quality goods and services, you are able to contribute to society better than most poor people. you're basically doing something that is expected of you, which is a wonderful trait. i think the problem is that you should at least be able to be generous towards someone without expecting to be rewarded.
Hua and Wang's Counter: this is a great example of how hard it is to invest in low income housing. it's not like it's going to end up being worse for everyone. if you don't like it, you're going to have to worry about it. the rich don't want to pay for it because they do n't have to pay taxes. they aren't going to be able to do anything about it, they just don't want.

FIGURE 5.14: Example counter-arguments generated by our approach and by the approach of Hua and Wang (2019). The italicized premise segment was identified as the weak premise by our approach.

implementation details of our model and the baselines, then will give an overview of both the automatic and manual evaluation and their results.

Experiment Setup

Data Similar to Subsection 5.3.2, we use the ChangeMyView (CMV) dataset of Jo et al. (2020) and follow the same mapping: The title of a post represents an argument's *conclusion* and its body is the *premises*, while each comment is a *counter-argument*. To ensure our models are trained on high-quality counters, we select for each post the comment with the highest argumentative quality score predicted by the model proposed by Gretz et al. (2020a).

To study counter-argument generation for settings where the conclusion is not mentioned explicitly, we use only the post's body as input and the title as training output to learn to generate the conclusion. Since users might also restate their post's main point (the conclusion) inside their post, this allows us to study and evaluate the correlation between a model's effectiveness in generating good counter-arguments and the level of implicitness of the conclusion in the input.

The stance-based ranking component relies on a classifier that assesses the stance polarity between two statements. To train such a classifier, we use the

dataset of Syed et al. (2021), which is based on the *Kialo.org* platform, where claims on controversial topics contributed by humans are organized in a hierarchical structure with supporting and opposing relations. We transformed the data into pairs of claims labeled as *pro* or *con*, and we split it by debates into 95.6k instances for training, 7.7k for validation, and 22.4k for testing.

Models We used BART as our base model (Lewis et al., 2020) to text generation and fine-tuned it starting from the *BART-large* checkpoint. We trained for three epochs using a learning rate of $5^e - 5$ and a batch size of 8. We then selected the checkpoint with the lowest error on the validation set. To find the best parameters α_a and α_b for the *Joint Two-decoders* model, we explored pairs of values between 0.1 and 1.0 on a sample of the training set and took the pair that led to the lowest validation loss: $\alpha_a = 0.7$ and $\alpha_b = 0.3$. To obtain a diverse set of candidate counters for ranking, we used nucleus sampling (Holtzman et al., 2019) with $p = 0.95$ and $top\_k = 50$. For the *Joint One-seq* model, we obtained relevant Wikipedia concepts from the input premises using Project Debater's API[8] that we used to prompt the output sequence (conclusion and counter-argument) to encourage diversity. As for the stance classifier, we fine-tuned *roberta-large* on the Kialo pairs for three epochs with learning rate $2e^{-5}$ and batch size 64. The trained classifier achieved an F1-score of 0.81 on the test split. To test its performance on the ChangeMyView data, we took a sample of 2k instances with pro pairs (an argument and its conclusion) and con pairs (conclusion and counter). The trained classifier resulted in an F_1-score of 0.70.

Baselines To study how effective transformer-based models are when the conclusion is not explicitly stated, we compare against four BART-based models, all trained on the conclusion and premises as input and the counter-argument as output but treated differently in the inference time.

In particular, the first baseline (*BART w/o conclusion*) relies only on the premises at inference time. To account for the missing conclusion, the second (*Pipeline-based*) generates a conclusion using another BART-based conclusion generation model trained independently on the training split of the CMV dataset. This can be seen as a pipeline alternative to our approach since conclusions and counters are learned independently. We also evaluate a variation of this pipeline approach that chooses the best counter among a diverse set of candidates using our ranking component (*Pipeline-based w/ Stance*). Finally, the fourth model is an oracle that knows the ground-truth conclusion in addition to the premises (*BART w/ conclusion*). Additionally, we compare our approach with the argument-undermining approach from the previous section in which the argument's weak points are first

[8]https://github.com/IBM/debater-eap-tutorial

Approach	BLEU	Be.F_1	Stance	Contr.
BART-based w/o Conclusion	0.149	0.138	0.814	0.447
Pipeline-based	0.148	0.142	0.816	0.437
Pipeline-based w/ Stance	0.141	0.142	0.852	0.615
Joint One-seq	0.143	***0.159**	0.850	*0.480
Joint One-seq w/ Stance	0.140	*0.147	**0.889**	***0.661**
Joint Two-decoders	*0.154	*0.148	0.798	0.423
Joint Two-decoders w/ Stance	***0.164**	*0.153	0.825	*0.652
BART-based w/ Conclusion	0.175	0.160	0.773	0.584
Argument Undermining	0.072	0.090	0.805	0.664

TABLE 5.10: Automatic evaluation of our two models, with and without *stance* ranking, compared to baselines, in terms of the similarity of the generated and the ground-truth counters (BLEU and BERT F_1-score) and of the counter's correct (opposing) stance. Stance is computed once using Project Debater's API (*Stance*) and once with our stance classifier (*Contrastiveness*). Results highlighted with * are significantly better than BART-based w/o Conclusion at a confidence level of 95%.

identified subject to its conclusion. Then a counter is generated to attack the weakest point(s).

Automatic Evaluation

In the following, we introduce the automatic evaluation measures used in our experiments. We then present the evaluation results of our approaches, as well as a detailed analysis of their effectiveness with respect to argument length (measured by the number of tokens) and conclusion implicitness.

Evaluation Measures To approximate the similarity of generated and ground-truth counters, we compute BLEU and BERT F_1-score. For each instance, we compare against all ground-truth counters and take the maximum score achieved. In addition, we measure the stance correctness of the generated counter with respect to the ground-truth conclusion in two ways: First, a *contrastiveness* score is computed using the stance classifier trained for our ranking component. It represents the average likelihood of classifying the counter and the corresponding ground-truth conclusion as *con* across the evaluation dataset. Second, a target-based *stance* score that measures the stance of both the conclusion and the counter towards the conclusion target. Given the validation set, we extract the target of each conclusion using our trained target inference model from Section 5.1.1 and then use Project Debater's API[9] to classify the conclusion's stance and the generated counter's stance towards the extracted target. The final measure is the absolute

[9]Debater API, `https://early-access-program.debater.res.ibm.com/`

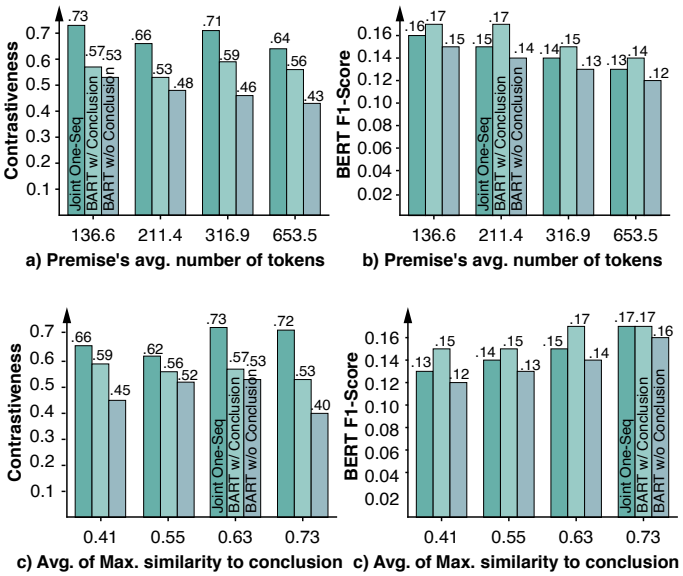

FIGURE 5.15: Contrastiveness and BERT F_1-scores of our approach *Joint One-seq* against the baseline subject to different levels of argument complexity (approximated by the number of tokens) and conclusion implicitness (approximated by the maximum similarity of the ground-truth conclusion to the premises). Illustration is taken from Alshomary and Wachsmuth (2023)

difference between the counter and conclusion scores, averaged across the evaluation dataset.

Results Table 5.10 shows the evaluation results. All approaches are close in BLEU and BERT F_1-score, with small but significant advantages for our models. We observe that the absence of explicit mention of the conclusion in the input (*BART w/o Conclusion*) worsens the results across all measures but the Stance score, and vice versa when introducing the conclusion (*BART w/ Conclusion*). This clearly indicates the importance of the conclusion in the process of counter-argument generation.

When the conclusion is not mentioned explicitly but has to be inferred, we can see that both our generation models, which jointly generate conclusions and counters, outperform the baselines in terms of correct stance. As expected, adding the ranking component to our approaches and the pipeline baseline consistently boosts the correctness, the best being *Joint One-seq w/ Stance* with stance score 0.889 and contrastiveness score of 0.661.

Although the argument undermining approach requires an explicit mention of the conclusion to rank premises according to their attackability, its effectiveness lacks behind. This could be because their model is trained on only a subset of the training data where the comments counter specific points in the post.

Analysis As discussed above, conclusions may appear in arguments implicitly, which we expect to correlate with the quality of the generated counters: the more explicit the conclusion, the better the generated counters. Moreover, we hypothesize that the longer an argument is, the more important the inference of its conclusion is for effective counter-argument generation.

We empirically investigate these two hypotheses by comparing the performance of the counter-argument generation models subject to *argument length* (in terms of the number of tokens) and to the degree of *conclusion implicitness* (in terms of the maximum similarity between the ground-truth conclusion and premises). In particular, for both dimensions, we sorted a sample of 2k instances from the test set accordingly and split it into five subsets of equal size. We then compare the BERT F_1-score and contrastiveness score of *Joint One-seq* against *BART w/o Conclusion* and *BART w/ Conclusion* on the respective subset.

Figure 5.15 shows the scores for all three models at different levels of argument length and conclusion implicitness. In Figure 5.15a, we see that the baseline's contrastiveness drops from 0.53 to 0.43 the longer the argument gets, while the scores for *BART w/ Conclusion* fluctuate relatively around 0.57. In contrast, our approach achieves scores between 0.64 and 0.73, indicating the benefit of the explicit modeling of conclusions. Figure 5.15c suggests that the more direct the conclusion is formulated in the premises, the better *BART w/o Conclusion*'s contrastiveness score gets, and vice versa for *BART w/ Conclusion* model.

We observe an unexpected drop in scores for arguments where conclusions have an average similarity of 0.7 to the premises. Upon inspection, we found that the baselines tend to copy parts of the premises with slight rephrasing. However, our approach, *Joint One-seq*, maintains high scores and also benefits from the clear formulation of the conclusion in the premises since this helps to generate better conclusions. Lastly, looking at BERT F_1-scores in Figures 5.15b and 5.15d, we notice that the values drop across all approaches as arguments get longer. Similarly, the more apparent the conclusion in the premises, the better the scores get.

Manual Evaluation

To gain more reliable insights into the performance of our approaches, we designed a human evaluation study to measure the quality of the generated counters in terms of relevance to the input argument and the correctness of their stance. Moreover, in a second study, we also let humans assess the validity of generated conclusions, in order to assess whether the multitask learning paradigm also boosts the performance of pre-trained language models over the single-task setting.

Counter-Arguments We selected 100 test set arguments randomly along with the counters generated by the two variations of our approach, *Joint One-seq w/ Stance* and *Joint Two-decoders w/ Stance*, as well as by two baselines, *BART w/o Conclu-*

	Annotator 1	Annotator 2	Annotator 3
Annotator 1	-	0.43	0.28
Annotator 2	0.43	-	0.30
Annotator 3	0.28	0.30	-

TABLE 5.11: Pairwise inter-annotator agreement in terms of Kendall's τ in the manual evaluation.

sion and *Pipeline-based*. Using the UpWork platform, we recruited three human annotators who are proficient in English with a job success of more than 90%. We presented them with the 100 arguments together with the texts of the four given counters, shuffled pseudo-randomly for each argument. For each argument, we then asked them to rank the texts based on their adequacy of being a counter-argument to the input argument, where we defined adequacy as follows:

> *An adequate counter is a text that (1) carries an argumentative and coherent language and (2) clearly represents an opposing stance to one of the main points in the input argument.*

Additionally, the annotators should provide comments describing their decision regarding the counters ranked first (the best) and fourth (the worst). Computing the inter-annotator agreement using Kendall's τ results in an average of 0.32 (ranging from 0.32 to 0.43), while we observed majority agreement on full ranks between the annotators in 78% of the evaluated cases.

Table 5.11 shows the pairwise inter-annotator agreement of the three annotators in terms of Kendall's τ, resulting in an average of 0.32, and ranging from 0.28 to 0.43. We observe that *Annotator 1* and *Annotator 2* agree notably more with each other than with *Annotator 3*. We observed a full-ranking majority agreement between our annotators in 78% of the evaluated cases.

Table 5.12 reports the mean of the average and majority ranks achieved by each approach. When considering cases with majority agreement, our model *Joint One-seq w/ Stance* performs best (mean rank 2.26). This also can be seen in Figure 5.16, where we plot the rank distribution for all approaches. In 55% of the cases, the approach generated counters that were ranked either first or second. However, the variation with two decoders falls short compared to all others (mean rank 2.72). This suggests that sharing only the encoder between the two tasks is not enough to generate relevant counters. Also, as indicated before, not being able to prompt the generated conclusions limits the diversity of candidates in the stance-based ranking component. Finally, we see that the *pipeline-based* baseline equipped with our ranking component is almost on par with our approaches, indicating the importance of promoting stance correctness.

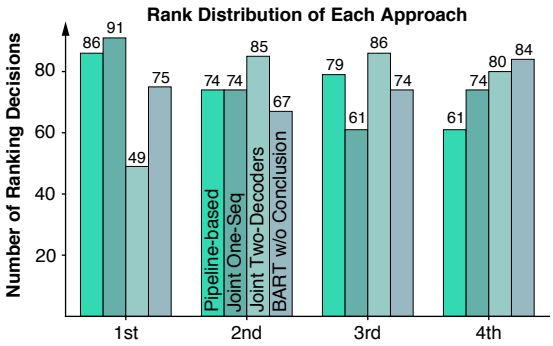

FIGURE 5.16: A histogram of the ranks that each of the manually evaluated approaches achieved on the 100 test cases, summing up the results of all three annotators. Illustration is taken from (Alshomary and Wachsmuth, 2023).

Counter Generation Approach	Average ↓	Majority ↓
BART-based w/o Conclusion	2.56	2.54
Pipeline-based w/ Stance	**2.38**	2.31
Joint One-seq w/ Stance	2.39	**2.26**
Joint Two-decoders w/ Stance	2.65	2.72

TABLE 5.12: Manual evaluation: The *average* and *majority* rank of the counters generated by our two approach variations and the two baselines. Lower is better.

Conclusions To investigate whether the joint learning of conclusion and counter-argument generation leads to more valid conclusions, we designed another human evaluation study, for which we defined validity in a simple way:

A conclusion is valid if humans can infer it from the input argument.

For 50 random arguments, we selected their ground-truth conclusion as well as two conclusions generated by the two variations of our approach and the best baseline (*Pipeline-based w/ Stance*), summing up to seven conclusions per argument. We hired two annotators through UpWork again. We asked them to read each argument and to evaluate the validity of each conclusion on a 3-point Likert scale, where 3 means that they strongly agree that the conclusion can be inferred and 1 means they strongly disagree. The agreement of the two annotators was 0.46 in terms of Cohen's κ.

Table 5.13 shows the average scores achieved by each evaluated model. With 1.42, the *pipeline-based* approach is notably worse than the others, indicating the advantage of multitask learning for conclusion and counter generation. The best score is achieved by *Joint Two-decoders w/ Stance* (2.03), being only 0.36 points below the ground-truth conclusion's score. Given the low effectiveness of this model on the counter-argument generation task, we assume that the training pro-

Conclusion Generation Approach	Validity ↑
Pipeline-based w/ Stance	1.42
Joint One-seq w/ Stance	1.91
Joint Two-decoders w/ Stance	2.03
Ground-truth Conclusions	**2.39**

TABLE 5.13: Manual evaluation: Average *validity* score from 1 (non-valid) to 3 (valid) of the conclusions generated by our two approach variations and by the baseline, in comparison to the ground truth.

cess optimized more towards generating conclusions, especially since the task may be easier than generating counters. A better weighting scheme for the two tasks may alleviate this in future work.

Finally, in the Appendix (Section A.3.3), we provide a qualitative analysis of an example argument and the corresponding counters generated by our approach and the baselines.

5.4 Concluding Remarks

To successfully counter the opponent's argument, skilled debaters need to understand the main point this argument is making as well as identify attackable premises on which it builds. Then, the argument can be countered by rebutting its main conclusion, undermining one of its weak premises, or undercutting by attacking the reasoning between its premises and the conclusion. In this chapter, we studied how to model this process. To this end, we proposed an approach of two main steps. The first analyzes the opponent's argument to identify its conclusion and weak premises. The second step uses these identified aspects to generate more relevant and effective counters. In the following, we will discuss our work's limitations and contributions to the overall picture of building an effective debate technology.

Limitations First, our approach builds on pre-trained language models (BART and GPT). Therefore, one should consider the challenges of generating faithful and unbiased text. Similar to the previous chapter, we suggest supporting our approach with other models to ensure faithful text generation, like claim verification (Wadden et al., 2020, Guo et al., 2022). Such models could ensure that the debate technology does not spit out false information hindering discussions.

Second, so far, we focused on inferring internal knowledge about the argument structure in our approach (Lauscher et al., 2022). However, other kinds of external knowledge, such as commonsense, can also benefit the task of counter-argument generation. Al-Khatib et al. (2020) constructed an argumentation knowledge graph comprising concepts with promote/suppress relations. Such a knowledge graph can be utilized for the task of countering arguments. For example, to counter an

argument supporting smoking, a counter to it can utilize the knowledge extracted from the graph about the promoting relationship between smoking and disease.

Third, in addressing the weak premise identification task, we assume that the argument's conclusion is given and learn the attackability scores accordingly. Nevertheless, as discussed throughout this thesis, conclusions are not necessarily explicitly mentioned in the argument. For such a scenario where conclusions are not given, one can use the inferred conclusions by our conclusion inference approach as a replacement. We should have evaluated the effectiveness of our approach with inferred conclusions, but we left this for future research.

Finally, another counter-argument we should have explored is attacking the reasoning between the premises and conclusions (argument undercut). In this regard, knowledge about argumentation schemes (Walton et al., 2008) used in the input argument can guide argument undercut. For example, knowing that an argument uses analogy as a scheme to draw parallels between two cases can inform a counter-argument generation model about the potential of attacking the validity of this analogy.

Contributions On the one hand, we bring the task of argument conclusion generation into the research focus by demonstrating its importance for other tasks, such as counter-argument generation. To this end, we clearly define the argument conclusion and propose a conceptual framework to infer it from an input argument. We show empirical evidence of the relation between premise and conclusion targets, and we introduce a method to learn to exploit this relation to infer conclusion targets from the input argument. Furthermore, we study the effectiveness of pretrained language models on this task. We found that these models learn to generate better conclusions when they are jointly learned along with the task of counter-argument generation.

On the other hand, we introduce a new model to predict weak premises that achieves state-of-the-art results. Instead of simply predicting the attackability of single premises, our model learns to rank all the premises of an argument according to their attackability with respect to the conclusion. We then present an approach that encodes the attackability of premises on the token level to generate the final counter that undermines the argument.

Overall, our methods to address the task of counter-argument generation can go hand in hand to form a single component in a debate technology responsible for generating effective counters to the opponent's argument. In this component, first, the opponent's argument gets analyzed by computing the weak scores of its premises and inferring its conclusion. Second, one of our counter-argument generation models (rebuttal or undermining) can generate the final counter accordingly. The empirical findings that demonstrate the effectiveness of our methods for gen-

erate counters also translate to an overall effectiveness of the debate technology in engaging in debates.

Chapter 6

Conclusion

This work aimed at the automatic generation of effective natural language arguments. To put our goal into perspective, we took debate technologies as an application scenario. In this scenario, given an audience and a controversial topic, an argumentation technology engages in a debate with a human to maximize audience agreement. To this end, we identified three research gaps where potential contributions could advance the effectiveness of these debate technologies. First, we argued that knowledge about the topic's discussion space is essential to guide the generation of relevant arguments (van Eemeren and Houtlosser, 1999). For this, we proposed methods to distill key points from a collection of retrieved arguments on the topic by modeling the key point candidacy on the sentence level. Second, we discuss the importance of the audience in argument generation. We proposed methods to learn the audience's belief system and used this knowledge to generate audience-aware arguments that are more effective. Third, we addressed the need for generating arguments that counters the opponent's argument. In this regard, we proposed methods to identify the main point and weak premises of the opponent's argument, and we then used this knowledge for counter-argument generation.

As highlighted in Figure 1.3, we envision a single framework composed of the proposed methods in our thesis. This framework first takes the topic and an audience representation and infers the corresponding key points (discussion space) and audience model. Second, to synthesize an argument on the given topic, the discussion space and the audience model are used to ensure the argument's relevancy to the topic and its effectiveness on the audience. Finally, our framework infers the opponent's argument conclusion and weak premises, and it then generates a corresponding counter-argument, using this inferred knowledge while maintaining relevance to the discussion space and effectiveness on the given audience. In the following, we will highlight our work's main findings and implications, as well as its limitations and future work.

6.1 Contribution and Main Findings

Throughout this work, our goal was the generation of effective arguments that increase agreement in the context of debate technologies. We identified three challenging areas where our research can help advance this goal. Accordingly, we conceptualized a debate technology framework consisting of three main components. The *identifying discussion space* model creates a set of talking key points representing the topic potential to ensure the relevance of generated arguments. The *modeling audience* component builds a model of the audience that can guide the generation of arguments to fit the audience's belief system to achieve better agreement. Finally, the *countering opponent's argument* component analyzes the input argument to infer its conclusion and weak premises to synthesize an appropriate counter. The following will summarize our main findings and contributions regarding each researched area and beyond.

Identifying Discussion Space We propose methods to model the key point candidacy on the sentence level by considering mainly the *argumentativeness* and *representativeness* criteria. In our experiments, we introduced *argument snippet generation* as a proxy task to assess the effectiveness of our key point candidacy models. We argued for this task's importance in the argument search context in providing an efficient overview of the main point of a retrieved argument. Our experiments indicate the limitation of general-purpose snippets and highlight the importance of modeling argumentativeness and centrality of sentences in producing better snippets of arguments. Moreover, to diversify the final set of key points, we studied how to model the sentence's contrastiveness as an extra criterion. In our snippet generation experiments, we presented methods to control the trade-off between the representativeness and contrastiveness of snippets based on the use case at hand. Finally, we provided empirical evidence of the effectiveness of our approach in generating key points by applying it to the key point analysis shared task proposed by Friedman et al. (2021a). In this regard, we also demonstrate the effectiveness of using contrastive learning to model the relation between arguments and key points. Among other submitted approaches, ours achieved the best results in generating relevant and representative key points and top effectiveness in matching these key points to arguments. Moreover, we compare our approach's generated key points to those generated by ChatGPT. The results demonstrate close effectiveness, with the advantage of our approach being more interpretable than ChatGPT. As mentioned, these key points form the topic potential that can guide the synthesis of arguments in a debate technology framework.

Modeling Knowledge about the Audience We are the first to propose and study the task of *audience-based argument generation*. We advocated the importance of addressing the audience's beliefs in argument generation to achieve agreement. To

this end, we introduced approaches to model and encode two audience representations: stance and moral-based. For the stance-based representation, we proposed two models that generate argumentative texts from scratch, one based on Seq2seq models and one based on pre-trained language models. On the other hand, our moral-based model builds on top of the project debater's API, guiding the generation of arguments to focus on a specific set of morals.

Overall, our experiments demonstrate the applicability of encoding these audience representations into generated arguments. When modeling the stances of users, we found that pre-trained language models generate more coherent texts than Seq2seq due to the pre-training process. Although our models were able to reflect the stance of users on specific big issues in the synthesized argument, we found that the relation between these big issues and the topic of discussion is crucial for successful encoding. For example, a model knowledgeable about the audience's stance on "the existence of aliens" could tell little about their potential belief regarding *abortion*. Furthermore, when modeling the audience's morals, we found that users rated arguments focusing on morals to be more effective than their general counterparts. In terms of audience, conservatives were more affected by the moral arguments than liberals. Finally, modeling the audience is an important step in a debate technology framework since it enables the generation of arguments that address the audience's beliefs – a crucial requirement to achieve agreement.

Modeling the Opponent's Argument To counter the opponent's argument, we argued that knowledge about its argumentative structure, such as its conclusion and weak premises, can help synthesize more relevant and effective counters. To this end, we proposed approaches to extracting and integrating such knowledge into the generation process. In particular, to infer the argument conclusion, we first proposed a model to infer its target given a set of premises by modeling their semantic relation using contrastive learning. We then study the effectiveness of pre-trained language models on this task and report their performance in single and multi-task settings. To identify weak premises, we proposed a ranking-based approach. We integrate this knowledge into the counter-argument generation process in the counter-generation component by implementing two LM-based models. The first implementation generates a counter that attacks one of the weak premises, while the other implementation generates a rebuttal considering the inferred conclusion.

Our experiments found that modeling the semantic relation between the premise and conclusion targets leads to more adequate conclusions. In modeling the weakness of premises, we found that this criterion is better learned as a ranking task relevant to the conclusion instead of a single premise classification. Finally, we demonstrate gain from modeling the conclusion and counter-argument generation tasks through a multitask approach. Our contribution enables debate technology

frameworks to generate counters to the opponent's argument that can either rebut its conclusion or attack its weak premises.

Beyond Debate Technologies Finally, we would like to point out that our research, although focused on debate technologies, also contributes to other computational argumentation (CA) areas. This thesis introduced tasks and corresponding approaches, such as the argument snippet and conclusion generation tasks. Research on the first task improves the usability of argument search engines through argument snippets that give an overview of the main points in the retrieved arguments, allowing users of argument search engines to assess the relevancy of each underlying argument quickly. On the other hand, studying the conclusion generation task contributes to argumentative writing support applications, for example, by helping students assess whether their premises sufficiently entail their intended conclusion Gurcke et al. (2021). Additionally, studying the identification of weak premises can help students write argumentative solid text by pointing out potential attackable claims in their text.

Moreover, beyond computational argumentation, our research demonstrates the importance of multitasking and contrastive learning as training paradigms in natural language processing. We successfully applied these learning paradigms to our tasks and showed a boost in performance as a result. In information retrieval, we discussed how users of search engines might have different search goals, which implies the need for domain-specific snippets. For example, in an argument search, the user is interested in a snippet highlighting the argument's main reasoning. Consequently, other search domains imply different specificity that is worth studying. Additionally, one can apply our key point analysis approach to summarizing other documents, such as online reviews. Here, one can also inject different modeling criteria instead of argumentativeness that better fit the domain at hand.

6.2 Limitations and Future Work

In this section, we present the potential limitations of our work and discuss future research directions to circumvent these limitations.

Inherited Limitations and Challenges Overall, our approaches build on and benefit from the rise of pre-trained language models. These benefits come with a cost. Typically, such technology inherits the same NLP challenges, such as bias and models' explainability. Due to its high social impact, gaining users' trust is crucial for a debate technology. Hence, empowering such technology with explainability and limiting its bias should be addressed. In general NLP settings, dedicated research lines have already been established to address these challenges (Danilevsky et al., 2020, Sheng et al., 2021). However, due to the subjective nature of argu-

mentative texts, studying such challenges with a debate scenario in focus can shed light on new emerging aspects of bias. Debate technologies also impact the user's autonomy due to their role in forming the user's opinion. If this debate technology fails to diversify its engagement with the user, it limits how informed their opinion is. Moreover, failing to identify misinformation could provide the user with false evidence. Thus, future research should investigate methods that ensure the diversity and trustworthiness of such technology (Li et al., 2022).

Approach Limitations As highlighted in Figure 1.3, we conceptualized an overall debate technology framework. Nevertheless, a few assumptions in our conceptual approach remained unsupported by empirical evidence and are open for future research. First, we hypothesized that the discussion space constructed through our key point modeling and aggregation approach could be used to guide the argument generation process. However, we did not perform experiments to this end. Future work can explore approaches similar to content planning and realization (Hua and Wang, 2019) where the generated key points for candidates to be selected from in the planning step.

The belief-based model learned about the audience can be used as a global parameter to guide the generation of counter-arguments. This direction requires the fusion of our methods for belief-based argument generation (Chapter 4) and counter-argument generation (Chapter 5). For example, the PPLM algorithm used in our belief-based claim generation approach can be applied on top of the counter-argument generation models at inference time to guide the generated counters towards a specific audience vocabulary. Finally, we discussed that the ultimate goal of our contribution is to build a joint representation of multiple audiences that can focus on the shared belief and use that to synthesize arguments that bring people together. We worked on building models of beliefs for a specific audience based on their stances or moral foundations. However, we did not address integrating models of beliefs of different audiences into one. All said, we can not make a concluding statement on the effectiveness of our conceptual framework in maximizing agreement in debate. Nevertheless, the empirical evidence that supports the effectiveness of the various components proposed throughout this thesis demonstrates potential success.

Moreover, as discussed in Chapter 3, our approach to generating discussion space focused on extractive summaries due to their intuitive interpretation, but this limits their applicability in cases where implicit reasoning is involved. In such situations, alternative approaches like inferring missing components or using abstractive summarization techniques should be considered. In addressing audience-aware argument generation (Chapter 4), we emphasize our methods' limitations and challenges. Firstly, generating arguments targeting a specific audience can have a societal impact, and transparency is crucial in ensuring the ethical use of

such technology. Privacy concerns should be addressed by adopting frameworks like the GDPR. Secondly, the models we considered for the audience in this thesis are reductionist and may not accurately capture individual beliefs and stances. Future work should explore more fine-grained approaches, such as learning latent representations directly from user texts. So far, we also relied on pre-trained language models (BART and GPT) to generate argumentative texts. In this regard, one should consider the challenges of generating faithful and unbiased text. We suggest supporting our approach with other models to ensure faithful text generation, like claim verification (Wadden et al., 2020, Guo et al., 2022). Finally, to counter-arguments, our focus was primarily on understanding the internal structure of arguments. However, additional external knowledge, such as commonsense, can be valuable for generating counter-arguments. As mentioned, Al-Khatib et al. (2020) developed an argumentation knowledge graph that we can utilize to counter arguments effectively.

Although our approaches, when evaluated against baselines, demonstrated strong results, their effectiveness is imperfect, and there is room for improvement. This limited effectiveness can result in failure cases with a cascading effect on the whole approach. For example, inferring a conclusion with the wrong stance will lead to generating wrong rebuttals. Wrongly identifying a weak premise can also lead to an ineffective counter. Similarly, when modeling the key point candidacy, we might extract nonrepresentative sentences from the collection as key points, which, in turn, leads to generating arguments that are not relevant to the discussion.

During the timeline of this work, the field of natural language processing witnessed much progress that rendered some of the models we built upon somewhat outdated. The biggest transformation in this field was the new era of large language models (LLMs), such as ChatGPT, that replaced typical language models like GPT-2 or BART. When possible, we tried to compare our models against ChatGPT (Chapter 3). We believe, however, that our findings still hold, and our approaches can still benefit from this emerging technology. One can envision implementing our approach on top of LLMs in many ways through prompting. For example, to re-implement our argument-undermining approach via LLM prompting, one can design a set of prompts that guide the LLM to investigate and find weak premises in a given argument to then use this information to generate the counter-argument. Such prompts can form a chain of thoughts that guide the LLM through the process, similar to the work of Wei et al. (2022). Overall, the various components of our debate framework can be re-implemented on top of LLMs to benefit from their generation power while ensuring a level of control over their generated text to guarantee levels of truthfulness and explainability.

Appendix A

Example Tables

In the following, we will present detailed tables of examples of the evaluated approaches throughout this thesis.

A.1 Modeling the Discussion Space

The follow-up experiment in Section 3.3.2 compared our approach to extracting key points (Ours) to ChatGPT on a set of 6 argument collections from the test split of Friedman et al. (2021a). Table A.1 presents the key points generated by each approach as well as the ground-truth for all of the six argument collections.

TABLE A.1: Examples of key points from our proposed approach. Only the top three key points are shown for brevity.

Topic: (t_1) Routine child vaccinations should be mandatory **- Stance:** Pro	
Ours	(1) Routine child vaccination should be mandatory so that the general population can thrive by being free of deadly diseases (2) Child vaccinations should be mandatory to provide decent health care to all (3) Child vaccinations should be mandatory so our children will be safe and protected
ChatGPT	(1) Routine child vaccinations should be mandatory to prevent the spread of diseases and protect children from deadly illnesses (2) Mandatory vaccinations ensure the health and safety of the general population, including vulnerable individuals such as infants and the elderly (3) Vaccinations save lives and prevent the unnecessary suffering caused by preventable diseases
GT.	(1) Routine child vaccinations are effective (2) Child vaccination saves lives (3) Routine child vaccinations are necessary to protect others

Continued on next page

Topic: (t_1) Routine child vaccinations should be mandatory **- Stance:** Con	
Ours	(1) Child vaccinations shouldn't be mandatory because some children can get sick or badly affected by vaccinations. (2) Vaccines contain heavy metals, they are toxic to children (3) The child population has a low degree of vulnerability, so vaccination is not urgent yet
ChatGPT	(1) Child vaccination should not be mandatory because of potential side effects and unknown risks (2) Parents should have the right to decide whether to vaccinate their children or not (3) Vaccines may not be necessary for children as they have stronger immune systems and may not be susceptible to the diseases
GT.	(1) Routine child vaccinations, or their side effects, are dangerous (2) Mandatory vaccination contradicts basic rights (3) The parents and not the state should decide
Topic: (t_2) Social media should be regulated by the government **- Stance:** Con	
Ours	(1) Regulation by government of social media platforms would be detrimental to free speech around the world (2) Government regulation of social media would be harmful to democracy (3) Social media platforms should not be regulated by the government because it is an invasion of privacy
ChatGPT	(1) Government regulation of social media platforms would be harmful to democracy and freedom of speech (2) Social media platforms should not be regulated by the government as it would infringe on privacy and freedom of expression (3) Social media platforms are private companies and should not be controlled by the government
GT.	(1) Social media regulation is not effective (2) Social media regulation harms privacy (3) Social media regulation harm freedom of speech and other democratic rights
Topic: (t_2) Social media should be regulated by the government **- Stance:** Pro	
Ours	(1) Social platforms must be regulated by governments to avoid hate crimes as well as political disinformation (2) Social media should be regulated by the government to prevent them from promoting dangerous activities and ideologies (3) Social media platforms should be regulated by the government to prevent bad-minded criminals from having an effect on others

Continued on next page

ChatGPT	(1) Social media platforms should be regulated by the government to prevent the spread of fake news and hoaxes (2) Regulation is necessary to protect society from crimes, hate speech, and harmful content on social media platforms (3) Government control of social media can help ensure online security and prevent the dissemination of false information
GT.	(1) Social media regulation protects the younger audiences (2) Social media regulation is necessary to deal with fake news (3) Social media regulation is beneficial to society at large

Continued on next page

Topic: (t_3) The USA is a good country to live in	**- Stance:** Pro
Ours	(1) The USA provides many benefits to their citizens that no other country has (2) The USA is a good country to live in as we have so many freedoms compared to other countries (3) The USA is a good country to live in because everyone has an equal opportunity to succeed
ChatGPT	(1) The United States is a country of opportunities and the American dream (2) The United States has a stable economy and good job opportunities (3) The United States offers a high quality of life and good healthcare
GT.	(1) The US offers great opportunities for individuals (2) The US has freedoms/ democratic rights (3) The US has a great environment/nature
Topic: (t_3) The USA is a good country to live in	**- Stance:** Con
Ours	(1) The poorest in society don't have access to either good health care or an adequate benefits system (2) The USA is not a good place to live in because of the wide variance between rich and poor (3) The USA is not a good place to live
ChatGPT	(1) High crime rates and lack of safety (2) High tax rates and expensive cost of living (3) Political divisions and social unrest
GT.	(1) The US has unfair health and education policies (2) The US has a problematic/divisive political system (3) The US has high taxation/high costs of living

TABLE A.1: Three example topics and the corresponding top three key points generated by our approach (Ours) and by ChatGPT, along with the ground-truth (GT.) key points

A.2 Modeling the Audience

This section will present various details of evaluated approaches for modeling the audience from Chapter 3.

A.2.1 Encoding Audience Model into Argument Generation

One of our approaches to modeling the audience's beliefs relies on their stances on big issues as a representation. As discussed in 4.3.2, not all big issues are relevant to the discussed topic. Table A.2 shows example big issues and their different

relatedness levels to example topics from our dataset as annotated by the authors of our work (Alshomary et al., 2021a). We consider the following levels:

- **Level 1** represents irrelevancy between the big issue and the topic
- **Level 2** reflects a slight correlation between the stance on the big issue and the stance on the topic
- **Level 3** represents more likelihood of correlation between the stances
- **Level 4** is for cases where the topic and the big issue are the same.

Relatedness	Big Issue	Topic
Level 1	Abortion	Do aliens exist?
Level 2	Death Penalty	Should Murder be Legalized
Level 3	Environmental Protection	Whaling
Level 4	Abortion	Is Abortion Wrong

TABLE A.2: Example Big Issues and topics for different relatedness levels between a big issues and topics in our dataset

A.2.2 Encoding Stances into Argumentative Claims

As presented in Section 4.3.1, we evaluate the ability of our approach to encode stances on big issues by computing the likelihood that the generated claims possess textual features reflecting the audience's beliefs as stances on big issues. We realize this by measuring the accuracy of predicting the audience's stances on big issues given the generated claims. We compute this accuracy for each big issue individually and report the results for all of them. In particular, we perform the following three steps for a given approach. First, we generate claims for all given users and topics in the test dataset. Second, we keep only instances in which users have a stance (pro/con) on the tested big issue and split the filtered dataset into training and testing. Finally, we train a simple TF-IDF-based linear classifier on the training set to predict the stance on the big issue given the text of the claim. The accuracy of the classifier on the test split then quantifies the likelihood of the generated claims possessing textual features that reflect the stance on the corresponding big issue. In Table A.3, we show the accuracy of the linear stance classifier when trained on claims generated by all approaches and tested against all 48 big issues.

Due to space restrictions, we abbreviated the big issues as follows: (AB) Abortion, (DL) Drug Legalization, GW (Global Warming), (DP) Death Penalty, (EP) Environmental Protection, (MM) Medical Marijuana, (AA) Affirmative Action,

(AR) Animal Rights, (BO) Barack Obama, (Cap) Capitalism, (CU) Civil Unions, (EC) Electoral College, (ET) Estate Tax, (EU) European Union, , (Euth) Euthanasia, (FR) Federal Reserve, (FT) Flat Tax, (FreeT) Free Trade, (GWE) Global Warming Exists, (Glob) Globalization, (GS) Gold Standard, (GR) Gun Rights, (HS) Homeschooling, (IC) Internet Censorship, (IIW) Iran-Iraq War, (LU) Labor Union, (LP) Legalized Prostitution, (M&M) Medicaid & Medicare, (MM) Medical Marijuana, (MI) Military Intervention, (MW) Minimum Wage, (NHC) National Health Care, (NRST) National Retail Sales Tax, (OM) Occupy Movement, (PT) Progressive Tax, (RP) Racial Profiling, (Red) Redistribution, (SP) Social Programs, (SS) Social Security, (Soc) Socialism, (StS) Stimulus Spending, (TL) Term Limits, (Tor) Torture, (UN) United Nations, (WA) War in Afghanistan, (WT) War on Terror, (Wel) Welfare.

A.2.3 The Moral-based Evaluation

In Section 4.3.4, we presented our user study to evaluate the morally-framed arguments by two different audiences, liberals and conservatives. In this study, we additionally asked our users to fill out a follow-up questionnaire to investigate their judgments of effectiveness in both challenging and empowering arguments. In particular, we asked our six annotators the following four questions regarding the challenging arguments:

- **Your views**: When arguments contested your stance on the topic, which of the following arguments did you see as more effective:
    1. Arguments that matched your views
    2. Arguments that challenged your views
    3. Neither of those was important
- **Your knowledge**: When arguments contested your stance on the topic, which of the following arguments did you see as more effective:
    1. Arguments based on views you already knew about
    2. Arguments that introduce views you were not familiar with
    3. Neither of those was important
- **Others' views**: When arguments contested your stance on the topic, which of the following arguments did you see as more effective:
    1. Arguments you saw as particularly convincing to people that share your views
    2. Arguments you saw as particularly convincing to people that rather oppose your views

3. Neither of those was important

- **Effectiveness**: When arguments contested your stance on the topic, which of the above three was most important for you to judge about effectiveness:

1. Your views
2. Your knowledge
3. Others' views

Similarly, we ask the same questions in the case of empowering arguments. Table A.4 and A.5 summarize the results of the four asked questions in both empowering and challenging arguments. In general, among the three aspects, the knowledge aspect was the most relevant one to the majority of the annotators (last row). When the annotators were asked about their own views (first row), conservatives favored mostly arguments that challenged their own views, while liberals favored mostly empowering arguments that matched their views. Regarding others' views, most conservatives and liberals valued empowering arguments that were particularly convincing to people that opposed their views. Nevertheless, we acknowledge the limited reliability of such self-assessment of one's moral judgments due to the complicated cognitive mechanisms behind it (Pizarro, 2000).

A.2.4 Moral-based Argument-Generation

We present in Table A.6 a sample of generated arguments on three topics focusing on both individualizing and binding morals, as well as uncontrolled.

TABLE A.6: Example generated arguments supporting *Affirmative Actions* for different focused morals.

Topic: Affirmative Actions
Binding argument: Law, students, policy and women are the four issues the crowd elaborated on. Let's explore the issue of law. The court found that the University of Michigan's Law School's affirmative action admission policies were promoting diversity within its school. Liberals have also argued for affirmative action to increase the representation of women and minorities among law students and law faculty. Following the enactment of Civil Rights laws in the 1960s, all educational institutions in the United States that receive federal funding have undertaken affirmative action to increase their racial diversity. The next issue is students. Another controversial decision of the Rehnquist court in 2003 was Grutter v. Bollinger which upheld affirmative action. It was in essence an affirmative action scheme to assist geographically disadvantaged students to gain tertiary education. The obvious solution to all the Affirmative Action controversy is to offer full financial assistance to all university students with need, regardless of race. Moving on to policy. This executive order created a National Women's Business Enterprise Policy and required government agencies to take affirmative action in support of women's business enterprises. Following the riots, the Malaysian Government introduced affirmative action policies to help the Bumiputera to achieve a higher economic quality of life than the Chinese. Ethiopian government policy has supported affirmative action for women since its inception in 1994. The ACLU's official position statements, as of January 2012, included the following policies: Affirmative action - The ACLU supports affirmative action. The last issue mentioned was women. Hunter is a supporter of affirmative action for women. He is highly in favor of affirmative action and supports setting aside funds for women and minorities. According to a poll taken by USA Today in 2005, the majority of Americans support affirmative action for women, while views on minority groups were more split. It support affirmative action for women. Affirmative Action and Impact Sourcing Tata BSS is an avid supporter of Affirmative Action and hence employs huge number of SC/ST community people in their operations thereby making a huge impact on the society.
Continued on next page

Individualizing argument: Employment, discrimination, universities and policy are the four issues the crowd elaborated on.
Let's explore the issue of employment. Affirmative action legislation has led to substantial improvements in the employment of minorities and women. They also require Federal contractors and subcontractors to take affirmative action to ensure equal employment opportunity in their employment processes. Historically and internationally, support for affirmative action has sought to achieve goals such as bridging inequalities in employment and pay, increasing access to education, promoting diversity, and redressing apparent past wrongs, harms, or hindrance.
The next issue is discrimination. The affirmative action program is designed to remedy the effects of past discrimination. In 2004, he reiterated his support, "I support affirmative action programs, including in appropriate instances consideration of race and gender in government contracting decisions, when the affirmative action program is designed to remedy the effects of past discrimination.". Alongside the prohibition against unfair discrimination, affirmative action is the second cornerstone of the EEA. A common approach to remedying discrimination is affirmative action.
Moving on to universities. The National Conference of State Legislatures held in Washington D.C. stated in a 2014 overview that many supporters for affirmative action argue that policies stemming from affirmative action help to open doors for historically excluded groups in workplace settings and higher education. Race-based affirmative action was necessary to achieve diversity and its educational benefits. Affirmative action provides some disadvantaged youth with increased chances of attending top-tier university. Action is thus more important than abstract knowledge.
The last issue mentioned was policy. According to its website, the NCP is committed to ideals of social justice, expressing support for affirmative action policies for the downtrodden members of society and for ensuring equal opportunities for all. Such legislation and affirmative action policies have been critical to bringing changes in societal attitudes. The school applies an affirmative action policy to ensure marginalized students from hardship semi-arid areas are given a chance. Affirmative Action Policy is to ensure the peace and stability in the pluralist society of Malaysia. It was found to address Malay grievances.

Continued on next page

Uncontrolled argument:
Discrimination, effectiveness, the supreme court and women are the four issues the crowd elaborated on.
Let's explore the issue of discrimination. 1961: Executive Order 10925: Required government contractors to "take affirmative action" to ensure non-discriminatory employment practices. It supports affirmative action as necessary in the fight for equality and it opposes all forms of racism and religious bigotry. Right to affirmative action All federal employers or federal contractors are required to take affirmative action to help counteract the effects of historical discrimination. CAMERA argues the Law of Return is justified under the Convention on the Elimination of All Forms of Racial Discrimination Article I, which CAMERA argues allows for affirmative action, because of the discrimination Jews faced during the Holocaust.
The next issue is effectiveness. In philosophy, Action is effective will. Collective action is the most effective means of preventing potential state and non-state aggressors from threatening other states. Such action is effective in spreading our message.
Moving on to the supreme court. Utter v. Bollinger: The Supreme Court of the United States upholds affirmative action in university admissions. The court found that the University of Michigan's Law School's affirmative action admission policies were promoting diversity within its school. Importantly, though, it ruled that a university was entitled to "no deference" on its judgment that race-based affirmative action was necessary to achieve diversity and its educational benefits. Another controversial decision of the Rehnquist court in 2003 was Grutter v. Bollinger which upheld affirmative action. A prominent advocate of affirmative action, he played a leading role in the twin Supreme Court cases-Grutter v Bollinger and Gratz v Bollinger-that upheld and clarified the importance of diversity as a compelling justification for affirmative action in higher education.
The last issue mentioned was women. Affirmative action programs to women in an effort to end gender discrimination. Gender-affirmative practices recognize and support an individual's unique gender self-identification and expression. Wesolowski supported affirmative action programs for Saskatoon Catholic teachers in 2001, arguing that past policies had discriminated against women and aboriginals. For women business owners, affirmative action programs are considered legislation that encourage government agencies and contractors to do business with women-owned businesses.

Topic: Legalizing Prostitution

Continued on next page

Binding argument:
These are the four issues the crowd addressed: the netherlands, adulthood, government and drugs.
Starting with the netherlands. Prostitution is legal and regulated in the Netherlands. In Switzerland, it has been legal since 1942. As a result of such views on prostitution, countries such as Germany, the Netherlands and New Zealand have fully legalized prostitution.
Adulthood was also mentioned. Prostitution of adults is legal in 10 rural counties in Nevada. Prostitution is a consensual sex act between adults and a victimless crime, thus the government should not prohibit this practice. The impugned Criminal Code provisions, by criminalizing many of the activities surrounding prostitution, adversely affect a great number of women. Prostitution was legal, as long as the women paid their license fees. It should not be illegal because the customers are satisfied.
Turning to government. Prostitution should be decriminalized, and as a libertarian insists the sex trade should not be regulated by the government. According to data from the Office for National Statistics, it contributed 5.3 billion to the UK economy in 2009. This, the sex work perspective asserts, will allow prostitution to be regulated by governments and business codes, protect sex trade workers, and improve the ability to prosecute people who hurt them. In 2005 the government also drafted the Sex Regulation Act which sought to further legalise and regulate prostitution. According to a Portuguese Government spokesperson, "The Government's opinion was that prostitution was not a crime.
The last issue mentioned was drugs. Mick Jagger has called for all drugs to be legalised on the Isle of Man. The Act legalised prostitution and put the women involved under police and medical control. In 2012, newly elected Guatemalan president Otto Perez Molina argued that all drugs should be legalized while attending the United Nations. Because of this, the Austrian Federal Ministry of the Interior wants to transform illegal prostitution into legal regulated prostitution.

Continued on next page

Individualizing argument: These are the four issues the crowd addressed: regulation, gambling, rape and crime.
Starting with regulation. According to proponents of regulation, prostitution should be considered a legitimate activity, which must be recognized and regulated, in order to protect the workers' rights and to prevent abuse. It should be legalised so it could be controlled safety. Maxine is a strong supporter of legalised brothels, more regulations around employment in the sex industry and increased sentences for perpetrators of violent crime. Barbara Brents and Kathryn Hausbeck state in their study that the legalization of prostitution in Nevada's brothels allows for improved regulation and protection for both businesses and workers.
Gambling was also mentioned. D'Amato believed that legalized gambling would be good for both the city and for those with businesses related to the entertainment industry. Maher favors ending corporate welfare and federal funding of non-profits as well as the legalization of gambling, prostitution, and cannabis.
Turning to rape. Kimberly Kay Hoang, assistant professor of sociology at the University of Chicago, who conducted a 2011 study of prostitutes in Ho Chi Minh City is quoted as saying "Legalising prostitution would also reduce violence and sex crimes such as rape and sexual violence. Legalising prostitution would also reduce violence and sex crimes such as rape and sexual violence. In 2006, the National Assembly legalized abortion care in cases of rape, as women regularly faced sexual violence, rape, and gang rape in the war-ravaged country. Legalizing brothels would make prostitution safer for women because it would allow prostitution to take place indoors.
The last issue mentioned was crime. Proposition K would decriminalize prostitution, enforce laws against crimes on sex workers, and disclose all investigations and prosecutions of violent crimes against sex workers. A May 1990 Australian Institute of Criminology report recommended that prostitution not be a criminal offence, since the laws were ineffective and endangered sex workers. In 1908, the king passed laws to legalize prostitution and help sex workers get medical care.

Continued on next page

Uncontrolled argument:
These are the four issues the crowd addressed: offices, sex workers, drugs and rural.
Starting with offices. Citizens' Assembly's Vanadzor Office president, Artur Sakunts, called for prostitution to be legalised and regulated. Prostitution in Hungary has been legalized and regulated by the government since 1999. In several countries, lotteries are legalized by the governments themselves. In response to the 1995 Federal-Provincial-Territorial Working Group on Prostitution report "Dealing with Prostitution in Canada," Toronto's Board of Health advocated decriminalisation in 1995, with the City taking the responsibility of regulating the industry.
Sex workers was also mentioned. The sex workers organisation "Guyana Sex Worker Coalition" and several NGOs called for prostitution to be legalised and regularization of sex work. Some sex-positive feminists believe that women and men can have positive experiences as sex workers and that where it is illegal, prostitution should be decriminalized. Since the mid-1970s, sex workers across the world have organised, demanding the decriminalisation of prostitution, equal protection under the law, improved working conditions, the right to pay taxes, travel and receive social benefits such as pensions. The sex work perspective maintains that prostitution is a legitimate form of work for women faced with the option of other bad jobs, therefore women ought to have the right to work in the sex trade free of prosecution or the fear of it.
Turning to drugs. If they did, prostitution and drugs would be legal. The Act legalised prostitution and put the women involved under police and medical control. In 2012, newly elected Guatemalan president Otto Perez Molina argued that all drugs should be legalized while attending the United Nations. He has studied the effects of drug criminalization for 15 years, and argues that all drugs should be legalized.
The last issue mentioned was rural. Prostitution of adults is legal in 11 rural counties in Nevada. Mayor Goodman supports legalizing prostitution in the city's downtown area as a revenue generator and tool for revitalization, although a majority of Nevadans polled in 2003 opposed the mayor's position.

TABLE A.6: Example generated arguments supporting *Affirmative Actions* for different focused morals.

A.3 Modeling the Opponent's Argument

This section will present various details of evaluated approaches for modeling the opponent's argument from Chapter 4.

A.3.1 Target Inference in Conclusion Generation

Table **??** introduces a set of argument examples of which we highlight the premise targets identified in the argument and the corresponding conclusion target that was inferred with each of our approaches, along with the ground-truth conclusions.

A.3.2 Argument Undermining

Table A.8 presents example counters generated by our argument-undermining approach and the one of Hua and Wang (2019).

TABLE A.8: Example generated arguments supporting *Affirmative Actions* for different focused morals.

Claim (title): it's better to have no freedom of speech than to be given the allusion of freedom of speech.
Premises (sentences): the reason why i believe this is that i like to know the rules first upfront , i hate people changing the rules in the middle of the game . the simplest comparison i can come up with is that if i am walking and then i accidentally walk into someone 's property than i get arrested for trespassing . i rather they straight up hang signs that say i am entering private property in the first place . *if i go my whole life believing i have free speech , then one day i get maced or tasered by a cop for speaking my mind* . i would 've expressed my opinions in another way . these countries where they say you have freedom but on the other hand oppress you are not free countries nor are they honest countries . in my view , they are worse than an openly oppressive country .
Our Counter: can you expand on what you mean here? this is simply demonstrably false. we have free speech. people have free speech in these countries because we are free people. i'm not sure how you got from you that you dislike freedom, but there's no contradiction. i am not free of speech.
Hua and Wang's Counter: if you believe in the right to free speech , then it is not a right . if you do n't agree with what you say , then you have to accept that it is a slippery slope . it 's not that you are n't allowed to say whatever you want , it 's that you do not have the right , and that you should not be allowed to speak freely . you can not have a right to express your views , but that does n't mean you should n't do it . if you
Continued on next page

Claim (title): i think hijab is good for the society and for women.
Premises (sentences): but i think that hijab is a good approach to reduce rape and harassment . *i think one of the reasons for rape are 1) not having a sexual partner 2) pornography , so here comes the role of the hijab so if women wear hijab rapists will not view her as a sexual object* . here is a video that shows a social experiment of the same woman wearing different cloth on the same streets of nyc :[– url –) . i want people that view hijab as some sort of backward way of dress to tell me why do you think like that ? and if possible change my view . (edit) .... . i agree with many of you guys that said rape is about power , but what i meant in my question is sexual harassment in general , by that i mean touching or grabbing private parts i could n't explain at first because of my english . so if touching and grabbing is under the crime of sexual harassment , do n't you think hijab will reduce those assaults ? so in short my point of view has changed when you guys said that rape is about power , i know think that a rapist will rape anyone regardless his cloth , but that does n't s include other sexual harassment means . excuse my english .
Our Counter: this point you're making is'why do you think women are raped if you'd like to do x with them?'i'm not sure what you're trying to get at here. are you saying people shouldn't think rape is a'bad idea '? because that's the most basic and fundamental reason a person should wear a hijab, and unless you don't believe in the concept of a'good idea ', you shouldn't think women are sexually harassed. rape is, in general, about power, sexual harassment, by the very definition of harassment.
Hua and Wang's Counter: there is no such thing as a " feminist " movement . it 's not just a matter of what it is , it 's a group of people who are in favor of it . women are not the only ones who are oppressed . women have been subject to the same rules as men . women are
Continued on next page

Claim (title): there is no logical way to prove that being unfair is wrong .
Premises (sentences): . one response was that " the party did n't deserve that type of treatment . " but that 's just another phrasing of being unfair . i decided to think for myself logically why being unfair is wrong and so far i 've only managed to come up with a few flawed answers . firstly , being unfair is self-evidently wrong . now this works out , until you realise that different people in different environments would find different values to be self-evident . *for example , if a human were to grow up alone without being in contact with another human since he was first conscious , then what he would n't find fairness to be self-evident* . instead , what he wo n't hesitate to do is kill others for whatever reasons he sees fit . he would see what he does as being acceptable , but we would n't . however , it would be impossible to convince him that others have a right to life because he grew not knowing empathy . if we apply this to the current context , then people in the west find different morals to be self-evident than people in asia or the middle east . yet everyone claims the other is inhumane , with no explanation how it is inhumane , or what is inhumane . another answer why being unfair is wrong is that it without fairness , society would n't function optimally . however if i purge the retired elderly or the ill who needlessly consume resources , then it would boost the cogs of society , wouldnt it ? its still considered wrong . therefore this answer is invalid . anyone have answers for the question " why is being unfair wrong ? "
Our Counter: how is being unfair any better than being wrong? fairness is subjective. in any society, fairness is subjective. if a person has a problem, does that mean their position is fair? the way we live the consequences of their decision means we can't change them. but why is that wrong?
Hua and Wang's Counter: i think it 's important to distinguish between the two scenarios , and i think that it 's more important to understand what you mean by " different " . i think you 're correct , but i think it

TABLE A.8: A list of examples of counter-arguments generated by our approach and by the approach of Hua and Wang (2019). The italicized premise segment was identified as the weak premise by our approach.

A.3.3 Argument Rebuttal

We perform a manual evaluation of the counters generated by our argument rebuttal approach (*Joint One-seq w/ Stance*) and baselines. Table A.9 shows an example argument discussing *Artificial Intelligence* along with counters generated by the two baselines as well as by our approach *Joint One-seq w/ Stance*. *BART w/o Conclusion* rephrases sentences from the input argument without generating a proper

counter, possibly due to ignorance of the conclusion. While the *pipeline-based* baseline equipped with our ranking component generates a somehow relevant conclusion, its counter is still vague and doesn't clearly oppose the argument's stance. Finally, *Joint One-seq* infers a conclusion that addresses the main point of the input argument (*Scientific law*), and counter it by pointing out the difficulty of defining *intelligent* , making it hard to be measured.

Upon exploring annotators' comments that justified their decisions of what is the best/worse counter, we identified some patterns. For example, *Joint One-seq* was most appreciated, because it generated argumentative and coherent counters that sometimes offered new perspectives. In contrast, the cases in which the model's output was ranked worst happen mainly due to being vague, incoherent, or diverging from the main topic. The counters of *BART w/o Conclusion* were ranked worse due to coherences sometimes, but often due to not opposing to the input argument.

Approach	Ab	AA	AR	BO	BF	Cap	CU	DP	DL	EC
GT	0.49	0.46	0.48	0.52	0.62	0.44	0.49	0.59	0.55	0.42
S2S-baseline	0.49	0.46	**0.66**	0.52	0.46	0.53	**0.56**	0.48	0.45	**0.54**
S2S-model	0.55	0.5	0.51	0.59	**0.52**	0.5	0.5	**0.55**	0.45	**0.54**
LM-baseline	0.48	0.51	0.55	0.52	0.46	0.49	0.48	0.5	0.49	0.41
LM-model	**0.58**	**0.6**	0.62	**0.6**	0.5	**0.58**	0.44	0.53	**0.56**	**0.54**
# Training instances	1610	1244	1806	1208	1092	1428	1288	1532	1538	1060
# Test instances	350	162	120	194	280	154	142	366	316	142

Approach	EP	ET	EU	Euth	FR	FT	FreeT	GM	GWE	Glob
GT	0.55	0.49	0.54	0.63	0.49	0.55	0.54	0.55	0.55	0.57
S2S-baseline	0.51	**0.57**	0.48	0.51	0.51	0.49	0.46	0.52	0.51	0.51
S2S-model	**0.58**	0.45	0.51	0.42	0.51	0.49	0.46	0.45	0.51	0.46
LM-baseline	0.56	0.45	**0.56**	**0.54**	0.51	**0.56**	**0.53**	**0.54**	0.54	**0.52**
LM-model	**0.58**	0.53	0.49	0.51	**0.55**	0.55	0.47	0.45	**0.61**	0.47
# Training instances	2196	1032	730	1380	622	922	1304	2098	1960	882
# Test instances	86	152	158	152	134	176	72	196	156	136

Approach	GS	GR	HS	IC	IIW	LU	LP	M&M	MM	MI
GT	0.52	0.48	0.51	0.55	0.53	0.51	0.55	0.49	0.5	0.5
S2S-baseline	0.54	0.55	0.51	0.41	0.4	0.41	0.49	0.44	0.57	0.54
S2S-model	0.42	0.48	0.54	0.49	0.49	0.4	0.47	0.55	0.57	0.49
LM-baseline	0.54	0.49	0.45	**0.51**	0.63	**0.49**	**0.6**	0.51	0.51	0.44
LM-model	**0.57**	**0.59**	**0.58**	0.42	**0.66**	0.47	0.52	**0.56**	**0.58**	**0.51**
# Training instances	594	1890	1302	1910	1668	1126	1186	1324	2096	910
# Test instances	130	182	138	98	68	146	230	138	138	144

Approach	MW	NHC	NRST	OM	PT	RP	Red	SB	SP	SS
GT	0.48	0.54	0.53	0.43	0.66	0.46	0.45	0.53	0.44	0.51
S2S-baseline	0.53	0.53	**0.54**	0.46	**0.51**	**0.61**	**0.55**	0.53	0.5	0.45
S2S-model	0.49	0.56	**0.54**	**0.54**	0.45	0.47	0.49	**0.53**	**0.54**	0.55
LM-baseline	0.59	0.57	**0.54**	0.43	0.46	0.54	0.49	0.45	0.41	0.48
LM-model	**0.65**	**0.62**	0.49	0.53	0.5	0.55	0.49	**0.53**	0.51	**0.56**
# Training instances	1580	1364	802	680	800	1728	764	1370	1278	1418
# Test instances	172	218	168	138	178	112	144	294	114	130

Approach	Soc	SS	TL	Tor.	UN	WA	WT	Wel.
GT	0.59	0.56	0.55	0.47	0.54	0.44	0.54	0.55
S2S-baseline	0.5	0.48	0.5	0.45	0.57	0.49	0.47	0.44
S2S-model	0.48	0.5	**0.54**	0.45	**0.62**	0.51	**0.52**	**0.55**
LM-baseline	0.47	0.4	0.41	0.51	0.59	0.45	**0.52**	0.49
LM-model	**0.59**	**0.63**	0.49	**0.53**	0.58	**0.57**	**0.52**	0.48
# Training instances	1100	664	1548	1616	1474	1564	1274	1256
# Test instances	174	120	112	152	160	136	252	196

TABLE A.3: Accuracy achieved by a stance classifier trained on claims generated by the evaluated models.

		Conservatives			Liberals		
		Empow.	**Chall.**	**All**	**Empow.**	**Chall.**	**All**
Knowledge	Know about	33.3%	0.0%	16.7%	33.3%	0.0%	16.7%
	Not familiar	**66.7%**	**66.7%**	**66.7%**	**66.7%**	**100.0%**	**83.3%**
	Neither	0.0%	33.3%	16.7%	0.0%	0.0%	0.0%
Own views	Matched	**33.3%**	0.0%	16.7%	**66.7%**	33.3%	**50.0%**
	Challenging	**33.3%**	**100.0%**	**66.7%**	33.7%	0.0%	16.7%
	Neither	**33.3%**	0.0%	16.7%	0.0%	**66.7%**	33.3%
Others' views	Share view	0.0%	0.0%	0.0%	33.3%	0.0%	16.7%
	Oppose view	**66.7%**	33.3%	**50.0%**	**66.7%**	**100.0%**	**83.3%**
	Neither	33.3%	**66.7%**	**50.0%**	0.0%	0.0%	0.0%
Effectiveness	Knowledge	**66.7%**	**66.7%**	**66.7%**	**100.0%**	**66.7%**	**83.3%**
	Views	33.3%	33.3%	33.3%	0.0%	33.3%	16.7%
	Neither	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%

TABLE A.4: Distribution of preferences (options) selected by the liberal and conservative annotators for each of the four asked questions for both the empowering and challenging cases.

		ALL		
		Empow.	**Chall.**	**All**
Knowledge	Know about	33.3%	0.0%	16.7%
	Not familiar	**66.7%**	**83.3%**	**75.0%**
	Neither	0.0%	16.7%	8.3%
Own views	Matched	**50.0%**	16.7%	33.3%
	Challenging	33.3%	**50.0%**	**41.7%**
	Neither	16.7%	33.3%	25.0%
Others' views	Share view	16.7%	0.0%	8.3%
	Oppose view	**66.7%**	**66.7%**	**66.7%**
	Neither	16.7%	33.3%	25.0%
Effectiveness	Knowledge	0.0%	0.0%	0.0%

TABLE A.5: Distribution of preferences (options) selected by both of the liberal and conservative annotators for each of the four asked questions for both the empowering and challenging cases.

Example 1

Argument: Relocating to the best universities is a budgetary concern , but also family and social relations concern for many people , which prevents all the best people from even applying to universities that would suit them the best . Online courses can recruit students from anywhere in the world much easier than traditional universities can because students do n't need to travel far away for the best education . This then ensures that universities have better access to the brightest people . For instance , Stanford University 's online course on Artificial Intelligence enabled people from 190 countries to join , and none of students receiving a score of 100 percent where from Stanford -LSB- 14 -RSB- . Improving the pool of students would automatically result in better academics , professionals and science , which would benefit the society better .

Ground truth conclusion: Online courses are a way to higher academic excellence

Premise Targets (ranked): Online courses

Target Embedding (learning): distance-learning

Example 2

Argument: Having a mobile phone helps us to learn in a lot of different ways . First we learn about technology ; about how to use the mobile phone . Second most phones today have apps -LRB- programs -RRB- to enable learning using the phone , or else through the internet . Phones can access online courses and lessons which can be provided in fun ways and can in some cases instantly tell you if you have the right answer . It may even sometimes be possible to do homework on a phone and send it to your teacher . Even without the internet phones can be used to provide short assignments , or to provide reminders to study .

Ground truth conclusion: Mobile phones help us to learn

Premise Targets (ranked): Phones

Target Embedding (learning): Mobile phones

Example 3

Argument: students who used to prepare Microsoft PowerPoint presentation , get an edge over others at an early stage of their career When children are allowed to play around with computer from a very early age , they get acquainted with the previously mentioned skills and become expert before facing professional world computers enable people to prepare presentations , draw complex graphs and pictures , document thesis in a simple though efficient way

Ground truth conclusion: it 's clear that computer has a positive effect on the children

Premise Targets (ranked): students who used to prepare Microsoft PowerPoint presentation for their school projects

Target Embedding (learning): future prospects of computers

TABLE A.7: Example arguments chosen from the test dataset, where premise targets and the conclusion target are highlighted in each argument. Along with that, we show the conclusion targets inferred by our approaches.

Conclusion (title): Artificial General Intelligence does not need to be conscious to be more intelligent or effective than humans **Premises (post):** In my opinion AGI needs the algorithms that make us intelligent and by having these missing pieces they will be a generally intelligent entity. I hear arguments that an AGI needs to have a subjective experience like a human being in order to be generally intelligent and I find this theory to be lackluster if the laws of physics can be computed than so can the algorithms that make general intelligence. After all we are information processing systems however, i am trying to explore all avenues of thought, even play devil's advocate and not get caught in some sort of echo chamber or even worse a spirit of the age.
BART-based w/o Conclusion: **Counter:** I don't think that AGI needs to have a subjective experience like a human being in order to be generally intelligent. I think it's more accurate to say that an AGI doesn't need to have subjective experience, but that doesn't mean that it can't have an objective experience.
Pipeline-based w/ Stance: **Conclusion:** i need to be more intelligent in order to be an AGI. **Counter:** intelligence is not determined by a subjective experience like a human being. it is determined by the laws of physics. an agi does not need to have an objective experience to be intelligent, but it does need to be able to understand the world around it.
Joint-based One-seq w/ Stance: **Conclusion:** Scientific law is the only thing that can make AGI generally intelligent. **Counter:** The problem with AGI is that we don't really know what it is that makes us intelligent. we have no idea how it works, what it's like to be an AGI, how it's different from a human being, or how it will work in the real world.

TABLE A.9: An example argument with conclusion (title) and premises (post), taken from *CMV*. Implicitly, the conclusion is also mentioned in the premises. Below, we show the output generated by each approach.

Bibliography

Yamen Ajjour, Wei-Fan Chen, Johannes Kiesel, Henning Wachsmuth, and Benno Stein. Unit Segmentation of Argumentative Texts. In *Proceedings of the Fourth Workshop on Argument Mining*. Association for Computational Linguistics, 2017.

Yamen Ajjour, Milad Alshomary, Henning Wachsmuth, and Benno Stein. Modeling Frames in Argumentation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing (EMNLP 2019)*, pages 2922–2932. ACL, November 2019a. URL `https://www.aclweb.org/anthology/D19-1290`.

Yamen Ajjour, Henning Wachsmuth, Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. Data Acquisition for Argument Search: The args.me corpus. In Christoph Benzmüller and Heiner Stuckenschmidt, editors, *42nd German Conference on Artificial Intelligence (KI 2019)*, pages 48–59, Berlin Heidelberg New York, September 2019b. Springer. doi:10.1007/978-3-030-30179-8_4.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649, 2018.

Khalid Al-Khatib, Henning Wachsmuth, Johannes Kiesel, Matthias Hagen, and Benno Stein. A News Editorial Corpus for Mining Argumentation Strategies. In Yuji Matsumoto and Rashmi Prasad, editors, *26th International Conference on Computational Linguistics (COLING 2016)*, pages 3433–3443. Association for Computational Linguistics, December 2016. URL `http://aclweb.org/anthology/C16-1324`.

Khalid Al-Khatib, Yufang Hou, Henning Wachsmuth, Charles Jochim, Francesca Bonin, and Benno Stein. End-to-end argumentation knowledge graph construction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7367–7374, 2020.

Khalid Al Khatib, Michael Völske, Shahbaz Syed, Nikolay Kolyada, and Benno Stein. Exploiting personal characteristics of debaters for predicting persuasiveness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7067–7072, 2020.

Khalid Al Khatib, Lukas Trautner, Henning Wachsmuth, Yufang Hou, and Benno Stein. Employing argumentation knowledge graphs for neural argument generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4744–4754, 2021.

Milad Alshomary and Henning Wachsmuth. Toward audience-aware argument generation. *Patterns*, 2(6):100253, 2021. ISSN 2666-3899. doi:https://doi.org/10.1016/j.patter.2021.100253. URL https://www.sciencedirect.com/science/article/pii/S2666389921000799.

Milad Alshomary and Henning Wachsmuth. Conclusion-based counter-argument generation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics, April 2023.

Milad Alshomary, Nick Düsterhus, and Henning Wachsmuth. Extractive snippet generation for arguments. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1969–1972, 2020a.

Milad Alshomary, Shahbaz Syed, Martin Potthast, and Henning Wachsmuth. Target inference in argument conclusion generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4334–4345, Online, July 2020b. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.399. URL https://www.aclweb.org/anthology/2020.acl-main.399.

Milad Alshomary, Wei-Fan Chen, Timon Gurcke, and Henning Wachsmuth. Belief-based generation of argumentative claims. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 224–233, Online, April 2021a. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2021.eacl-main.17.

Milad Alshomary, Timon Gurcke, Shahbaz Syed, Philipp Heinisch, Maximilian Spliethöver, Philipp Cimiano, Martin Potthast, and Henning Wachsmuth.

Key point analysis via contrastive learning and extractive argument summarization. In *Proceedings of the 8th Workshop on Argument Mining*, pages 184–189, Punta Cana, Dominican Republic, November 2021b. Association for Computational Linguistics. doi:10.18653/v1/2021.argmining-1.19. URL https://aclanthology.org/2021.argmining-1.19.

Milad Alshomary, Shahbaz Syed, Arkajit Dhar, Martin Potthast, and Henning Wachsmuth. Counter-argument generation by attacking weak premises. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1816–1827, 2021c.

Milad Alshomary, Roxanne El Baff, Timon Gurcke, and Henning Wachsmuth. The moral debater: A study on the computational generation of morally framed arguments. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8782–8797, 2022a.

Milad Alshomary, Jonas Rieskamp, and Henning Wachsmuth. Generating contrastive snippets for argument search. In *Computational Models of Argument*, pages 21–31. IOS Press, 2022b.

Oscar Araque, Lorenzo Gatti, and Kyriaki Kalimeri. Moralstrength: Exploiting a moral lexicon and embedding similarity for moral foundations prediction. *Knowledge-based systems*, 191:105184, 2020.

Aristotle and George A. Kennedy. *On Rhetoric: A Theory of Civic Discourse*. Oxford: Oxford University Press, 2006.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

Roy Bar-Haim, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. Stance Classification of Context-Dependent Claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 251–261. Association for Computational Linguistics, 2017. URL http://aclweb.org/anthology/E17-1024.

Roy Bar-Haim, Lilach Eden, Roni Friedman, Yoav Kantor, Dan Lahav, and Noam Slonim. From arguments to key points: Towards automatic argument summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4029–4039. Association for Computational Linguistics, July 2020a. doi:10.18653/v1/2020.acl-main.371. URL https://www.aclweb.org/anthology/2020.acl-main.371.

Roy Bar-Haim, Yoav Kantor, Lilach Eden, Roni Friedman, Dan Lahav, and Noam Slonim. Quantitative argument summarization and beyond: Cross-domain key point analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 39–49, Online, November 2020b. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.3. URL https://aclanthology.org/2020.emnlp-main.3.

Giannis Bekoulis, Christina Papagiannopoulou, and Nikos Deligiannis. A review on fact extraction and verification. *ACM Computing Surveys (CSUR)*, 55(1): 1–35, 2021.

Trevor J. M. Bench-Capon, Sylvie Doutre, and Paul E. Dunne. Value-based argumentation frameworks. In *Artificial Intelligence*, pages 444–453, 2002.

Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

Andrzej Białecki, Robert Muir, Grant Ingersoll, and Lucid Imagination. Apache lucene 4. In *SIGIR 2012 Workshop on Open-Source Information Retrieval*, 2012.

Yonatan Bilu, Daniel Hershcovich, and Noam Slonim. Automatic claim negation: Why, how and when. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 84–93, Denver, CO, June 2015. Association for Computational Linguistics. doi:10.3115/v1/W15-0511. URL https://www.aclweb.org/anthology/W15-0511.

Yonatan Bilu, Ariel Gera, Danel Hershcovich, Benjamin Sznajder, Dan Lahav, Guy Moshkowich, Anael Malet, Assaf Gavron, and Noam Slonim. Argument Invention from First Principles. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pages 1013–1026. Association for Computational Linguistics, 2019. URL https://www.aclweb.org/anthology/P19-1097/.

Umanga Bista, Alexander Mathews, Minjeong Shin, Aditya Krishna Menon, and Lexing Xie. Comparative document summarisation via classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 20–28, 2019.

Umanga Bista, Alexander Patrick Mathews, Aditya Krishna Menon, and Lexing Xie. Supmmd: A sentence importance model for extractive summarisation using maximum mean discrepancy. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4108–4122. Association for Computational Linguistics, 2020.

doi:10.18653/v1/2020.findings-emnlp.367. URL https://doi.org/10.18653/v1/2020.findings-emnlp.367.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

Filip Boltuzic and Jan Šnajder. Fill the gap! Analyzing implicit premises between claims from online debates. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 124–133. Association for Computational Linguistics, 2016. doi:10.18653/v1/W16-2815. URL https://www.aclweb.org/anthology/W16-2815.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. In *Advances in neural information processing systems*, pages 737–744, 1994.

Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. 2010.

Giuseppe Carenini and Johanna D Moore. Generating and evaluating evaluative arguments. *Artificial Intelligence*, 170(11):925–952, 2006.

Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty: 8th European Conference, ECSQARU 2005, Barcelona, Spain, July 6-8, 2005. Proceedings 8*, pages 378–389. Springer, 2005.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

Tuhin Chakrabarty, Christopher Hidey, and Kathleen Mckeown. Imho fine-tuning improves claim detection. 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 558–563, 2019a.

Tuhin Chakrabarty, Christopher Hidey, Smaranda Muresan, Kathy McKeown, and Alyssa Hwang. AMPERSAND: Argument mining for PERSuAsive oNline discussions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2933–2943, Hong Kong, China, November 2019b. Association for Computational Linguistics.

doi:10.18653/v1/D19-1291. URL https://aclanthology.org/D19-1291.

Tuhin Chakrabarty, Aadit Trivedi, and Smaranda Muresan. Implicit premise generation with discourse-aware commonsense knowledge models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6247–6252, 2021.

Wei-Fan Chen, Shahbaz Syed, Benno Stein, Matthias Hagen, and Martin Potthast. Abstractive snippet generation. In *Proceedings of The Web Conference 2020*, pages 1309–1319, 2020.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, 2014.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences*, 118(9):e2023301118, 2021. doi:10.1073/pnas.2023301118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2023301118.

Ronan Collobert. Word embeddings through hellinger pca. In *in Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. Citeseer, 2014.

Angelo Costa, Vicente Julian, and Paulo Novais. *Personal Assistants: Emerging Computational Technologies*, volume 132. Springer, 2017.

Bruce Croft, Donald Metzler, and Trevor Strohman. *Search Engines: Information Retrieval in Practice*. Addison-Wesley, USA, 1st edition, 2009.

Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. A survey of the state of explainable AI for natural language processing. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 447–459, Suzhou, China, December 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.aacl-main.46.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1edEyBKDS.

Johannes Daxenberger, Steffen Eger, Ivan Habernal, Christian Stab, and Iryna Gurevych. What is the Essence of a Claim? Cross-Domain Claim Identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 2045–2056. Association for Computational Linguistics, 2017. URL https://www.aclweb.org/anthology/D17-1218/.

Johannes Daxenberger, Benjamin Schiller, Chris Stahlhut, Erik Kaiser, and Iryna Gurevych. Argumentext: argument classification and clustering in a generalized search scenario. *Datenbank-Spektrum*, 20:115–121, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Emily Dinan, Stephen Roller, Kurt Shuster, Michael Auli, and Jason Weston. Knowledge-powered conversational agents. 2018.

Liat Ein Dor, Alon Halfon, Yoav Kantor, Ran Levy, Yosi Mass, Ruty Rinott, Eyal Shnarch, and Noam Slonim. Semantic relatedness of wikipedia concepts–benchmark data and a working solution. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

Jonas Dorsch and Henning Wachsmuth. Semi-supervised cleansing of web argument corpora. In *Proceedings of the 7th Workshop on Argument Mining*, pages 19–29, Online, December 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.argmining-1.3.

Lorik Dumani and Ralf Schenkel. A systematic comparison of methods for finding good premises for claims. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 957–960, 2019.

Phan Minh 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.

Esin Durmus and Claire Cardie. Exploring the role of prior beliefs for argument persuasion. In *Proceedings of the 2018 Conference of the North American Chap-*

ter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1035–1045, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi:10.18653/v1/N18-1094. URL https://www.aclweb.org/anthology/N18-1094.

Charlie Egan, Advaith Siddharthan, and Adam Wyner. Summarising the points made in online political debates. In *Proceedings of the 3rd Workshop on Argument Mining, The 54th Annual Meeting of the Association for Computational Linguistics*, pages 134–143. Association for Computational Linguistics (ACL), 2016.

Liat Ein-Dor, Eyal Shnarch, Lena Dankin, Alon Halfon, Benjamin Sznajder, Ariel Gera, Carlos Alzate, Martin Gleize, Leshem Choshen, Yufang Hou, et al. Corpus wide argument mining?a working solution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7683–7691, 2020.

Roxanne El Baff, Henning Wachsmuth, Khalid Al Khatib, and Benno Stein. Challenge or empower: Revisiting argumentation quality in a news editorial corpus. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 454–464, 2018.

Roxanne El Baff, Henning Wachsmuth, Khalid Al Khatib, Manfred Stede, and Benno Stein. Computational argumentation synthesis as a language modeling task. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 54–64, Tokyo, Japan, October–November 2019. Association for Computational Linguistics. doi:10.18653/v1/W19-8607. URL https://www.aclweb.org/anthology/W19-8607.

Roxanne El Baff, Khalid Al Khatib, Benno Stein, and Henning Wachsmuth. Persuasiveness of news editorials depending on ideology and personality. In *Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*, volume 3, pages 29–40. Association for Computational Linguistics, 2020.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, and Hoda Korashy Mohamed. Automatic text summarization: A comprehensive survey. *Expert Syst. Appl.*, 165:113679, 2021.

Günes Erkan and Dragomir R Radev. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22: 457–479, 2004.

Mohammad Hassan Falakmasir, Kevin D. Ashley, Christian D. Schunn, and Diane J. Litman. Identifying thesis and conclusion statements in student essays to scaffold peer review. In Stefan Trausan-Matu, Kristy Elizabeth Boyer, Martha E.

Crosby, and Kitty Panourgia, editors, *Intelligent Tutoring Systems - 12th International Conference, ITS 2014, Honolulu, HI, USA, June 5-9, 2014. Proceedings*, volume 8474 of *Lecture Notes in Computer Science*, pages 254–259. Springer, 2014. doi:10.1007/978-3-319-07221-0_31. URL https://doi.org/10.1007/978-3-319-07221-0_31.

Angela Fan, Mike , and Yann Dauphin. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, 2018.

Matthew Feinberg and Robb Willer. From gulf to bridge: When do moral arguments facilitate political influence? *Personality and Social Psychology Bulletin*, 41(12):1665–1681, 2015.

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

Roni Friedman, Lena Dankin, Yufang Hou, Ranit Aharonov, Yoav Katz, and Noam Slonim. Overview of the 2021 key point analysis shared task. In *Proceedings of the 8th Workshop on Argument Mining*, pages 154–164, Punta Cana, Dominican Republic, November 2021a. Association for Computational Linguistics. doi:10.18653/v1/2021.argmining-1.16. URL https://aclanthology.org/2021.argmining-1.16.

Roni Friedman, Lena Dankin, Yoav Katz, Yufang Hou, and Noam Slonim. Overview of KPA-2021 shared task: Key point based quantitative summarization, November 2021b.

Dean Fulgoni, Jordan Carpenter, Lyle Ungar, and Daniel Preoţiuc-Pietro. An empirical exploration of moral foundations theory in partisan news sources. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3730–3736, 2016.

Cristina Garbacea and Qiaozhu Mei. Neural language generation: Formulation, methods, and evaluation. *arXiv preprint arXiv:2007.15780*, 2020.

Saeedeh Gholamrezazadeh, Mohsen Amini Salehi, and Bahareh Gholamzadeh. A comprehensive survey on text summarization systems. In *Proceedings of the 2nd CSA*, pages 1–6, 2009.

Thomas F Gordon and Douglas 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.

Jesse Graham, Jonathan Haidt, and Brian A Nosek. Liberals and conservatives rely on different sets of moral foundations. *Journal of personality and social psychology*, 96(5):1029, 2009.

Floriana Grasso, Alison Cawsey, and Ray Jones. Dialectical argumentation to solve conflicts in advice giving: a case study in the promotion of healthy nutrition. *International Journal of Human-Computer Studies*, 53(6):1077–1115, 2000.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. Ieee, 2013.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Shai Gretz, Yonatan Bilu, Edo Cohen-Karlik, and Noam Slonim. The workweek is the best time to start a family – a study of GPT-2 based claim generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 528–544, Online, November 2020a. Association for Computational Linguistics. doi:10.18653/v1/2020.findings-emnlp.47. URL `https://www.aclweb.org/anthology/2020.findings-emnlp.47`.

Shai Gretz, Roni Friedman, Edo Cohen-Karlik, Assaf Toledo, Dan Lahav, Ranit Aharonov, and Noam Slonim. A large-scale dataset for argument quality ranking: Construction and analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7805–7813, 2020b.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206, 2022.

Timon Gurcke, Milad Alshomary, and Henning Wachsmuth. Assessing the sufficiency of arguments through conclusion generation. In *Proceedings of the 8th Workshop on Argument Mining*, pages 67–77, 2021.

Ivan Habernal and Iryna Gurevych. Exploiting Debate Portals for Semi-Supervised Argumentation Mining in User-Generated Web Discourse. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2127–2137. Association for Computational Linguistics, 2015. doi:10.18653/v1/D15-1255. URL `http://aclweb.org/anthology/D15-1255`.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. The argument reasoning comprehension task: Identification and reconstruction of

implicit warrants. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1930–1940, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi:10.18653/v1/N18-1175. URL https://www.aclweb.org/anthology/N18-1175.

Jonathan Haidt. *The righteous mind: Why good people are divided by politics and religion*. Vintage, 2012.

Jonathan Haidt and Craig Joseph. Intuitive ethics: How innately prepared intuitions generate culturally variable virtues. *Daedalus*, 133(4):55–66, 2004.

Shuguang Han, Xuanhui Wang, Mike Bendersky, and Marc Najork. Learning-to-rank with bert in tf-ranking. *ArXiv*, abs/2004.08476, 2020.

Pieter Sjoerd Hasper. Aristotle's sophistical refutations. a translation. In *Fallacious Arguments in Ancient Philosophy*, pages 13–54. Brill mentis, 2013.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *NIPS*, pages 1693–1701, 2015. URL http://papers.nips.cc/paper/5945-teaching-machines-to-read-and-comprehend.

Christopher Hidey and Kathleen McKeown. Fixed that for you: Generating contrastive claims with semantic edits. 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 1756–1767, 2019.

Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer, 2015.

Carolin Holtermann, Anne Lauscher, and Simone Paolo Ponzetto. Fair and argumentative language modeling for computational argumentation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7841–7861, 2022.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. Learning to write with cooperative discriminators. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1638–1649, 2018.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020. URL https://doi.org/10.5281/zenodo.1212303.

Joe Hoover, Gwenyth Portillo-Wightman, Leigh Yeh, Shreya Havaldar, Aida Mostafazadeh Davani, Ying Lin, Brendan Kennedy, Mohammad Atari, Zahra Kamel, Madelyn Mendlen, et al. Moral foundations twitter corpus: A collection of 35k tweets annotated for moral sentiment. *Social Psychological and Personality Science*, 11(8):1057–1071, 2020.

Sheng-Luan Hou, Xi-Kun Huang, Chao-Qun Fei, Shu-Han Zhang, Yang-Yang Li, Qi-Lin Sun, and Chuan-Qing Wang. A survey of text summarization approaches based on deep learning. *Journal of Computer Science and Technology*, 36(3): 633–663, 2021.

Xinyu Hua and Lu Wang. Neural argument generation augmented with externally retrieved evidence. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 219–230, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi:10.18653/v1/P18-1021. URL https://www.aclweb.org/anthology/P18-1021.

Xinyu Hua and Lu Wang. Sentence-level content planning and style specification for neural text generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 591–602, 2019.

Ioana Hulpus, Jonathan Kobbe, Heiner Stuckenschmidt, and Graeme Hirst. Knowledge graphs meet moral values. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 71–80, 2020.

Fattaneh Jabbari, Mohammad Hassan Falakmasir, and Kevin D. Ashley. Identifying thesis statements in student essays: The class imbalance challenge and resolution. In *Proceedings of the Twenty-Ninth International Florida Artificial Intelligence Research Society Conference*, pages 220–225, 2016. URL http://www.aaai.org/ocs/index.php/FLAIRS/FLAIRS16/paper/view/12971.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Yohan Jo, Seojin Bang, Emaad Manzoor, Eduard Hovy, and Chris Reed. Detecting attackable sentences in arguments. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1–23, Online, November 2020. Association for Computational Linguistics.

Daniel Jurafsky and James H Martin. *Speech and language processing*, volume 710. 2000.

Tapas Kanungo and David Orr. Predicting the readability of short web summaries. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 202–211, 2009.

Maurice G Kendall and B Babington Smith. The problem of m rankings. *The annals of mathematical statistics*, 10(3):275–287, 1939.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*, 2019.

Johannes Kiesel, Milad Alshomary, Nicolas Handke, Xiaoni Cai, Henning Wachsmuth, and Benno Stein. Identifying the human values behind arguments. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4459–4471, 2022.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P17-4012`.

Jonathan Kobbe, Ines Rehbein, Ioana Hulpu?, and Heiner Stuckenschmidt. Exploring morality in argumentation. In *Proceedings of the 7th Workshop on Argument Mining*, pages 30–40, 2020.

Sarawoot Kongyoung, Craig Macdonald, and Iadh Ounis. Multi-task learning using dynamic task weighting for conversational question answering. In *Proceedings of the 5th International Workshop on Search-Oriented Conversational AI (SCAI)*, pages 17–26, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.scai-1.3. URL `https://aclanthology.org/2020.scai-1.3`.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

Anne Lauscher, Henning Wachsmuth, Iryna Gurevych, and Goran Glavaš. Scientia potentia est–on the role of knowledge in computational argumentation. *Transactions of the Association for Computational Linguistics*, 10:1392–1422, 2022.

Ran Levy, Shai Gretz, Benjamin Sznajder, Shay Hummel, Ranit Aharonov, and Noam Slonim. Unsupervised corpus–wide claim detection. In *Proceedings of the 4th Workshop on Argument Mining*, pages 79–84, 2017.

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

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, 2020.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003, Berlin, Germany, August 2016. Association for Computational Linguistics. doi:10.18653/v1/P16-1094. URL `https://www.aclweb.org/anthology/P16-1094`.

Liangda Li, Ke Zhou, Gui-Rong Xue, Hongyuan Zha, and Yong Yu. Enhancing diversity, coverage and balance for summarization through structure learning. In *Proceedings of the 18th international conference on World wide web*, pages 71–80, 2009.

Wei Li, Wenhao Wu, Moye Chen, Jiachen Liu, Xinyan Xiao, and Hua Wu. Faithfulness in natural language generation: A systematic survey of analysis, evaluation and optimization methods. *arXiv preprint arXiv:2203.05227*, 2022.

Shao Fen Liang, Siobhan Devlin, and John Tait. Evaluating web search result summaries. In *Proceedings of the 28th ECIR*, pages 96–106. Springer, 2006.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

Chin-Yew Lin and Eduard Hovy. The automated acquisition of topic signatures for text summarization. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*, 2000. URL `https://www.aclweb.org/anthology/C00-1072`.

Yang Liu. Fine-tune BERT for extractive summarization. *arXiv preprint arXiv:1903.10318*, 2019.

Sidi Lu, Yaoming Zhu, Weinan Zhang, Jun Wang, and Yong Yu. Neural text generation: Past, present and beyond. *arXiv preprint arXiv:1803.07133*, 2018.

Stephanie M. Lukin, Pranav Anand, Marilyn A. Walker, and Steve Whittaker. Argument strength is in the eye of the beholder: Audience effects in persuasion. In *EACL*, 2017.

Mari-Carmen Marcos, Ferran Gavin, and Ioannis Arapakis. Effect of snippets on user experience in web search. In *Proceedings of the 16th HCI*, page 47, 2015.

Caleb Martin, Huichen Yang, and William Hsu. Kddie at semeval-2022 task 11: Using deberta for named entity recognition. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1531–1535, 2022.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL `http://arxiv.org/abs/1301.3781`.

Amita Misra, Brian Ecker, and Marilyn Walker. Measuring the similarity of sentential arguments in dialogue. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 276–287, Los Angeles, September 2016. Association for Computational Linguistics. URL `https://aclanthology.org/W16-3636`.

Alex J Novikoff. Toward a cultural history of scholastic disputation. *The American Historical Review*, 117(2):331–364, 2012.

Matan Orbach, Yonatan Bilu, Ariel Gera, Yoav Kantor, Lena Dankin, Tamar Lavee, Lili Kotlerman, Shachar Mirkin, Michal Jacovi, Ranit Aharonov, and Noam Slonim. A dataset of general-purpose rebuttal. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5591–5601, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1561. URL `https://www.aclweb.org/anthology/D19-1561`.

Matan Orbach, Yonatan Bilu, Assaf Toledo, Dan Lahav, Michal Jacovi, Ranit Aharonov, and Noam Slonim. Out of the echo chamber: Detecting countering debate speeches. In *Proceedings of the 58th Annual Meeting of the Association*

for Computational Linguistics, pages 7073–7086, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.633. URL https://www.aclweb.org/anthology/2020.acl-main.633.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

Rama Kumar Pasumarthi, Sebastian Bruch, Xuanhui Wang, Cheng Li, Michael Bendersky, Marc Najork, Jan Pfeifer, Nadav Golbandi, Rohan Anil, and Stephan Wolf. Tf-ranking: Scalable tensorflow library for learning-to-rank. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2970–2978, 2019.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 1532–1543. ACL, 2014. doi:10.3115/v1/d14-1162.

Chaim Perelman. The new rhetoric. In *Pragmatics of natural languages*, pages 145–149. Springer, 1971.

Georgios Petasis and Vangelis Karkaletsis. Identifying argument components through textrank. In *Proceedings of the Third Workshop on Argument Mining, hosted by the 54th Annual Meeting of the Association for Computational Linguistics, ArgMining@ACL 2016, August 12, Berlin, Germany*, 2016. URL https://www.aclweb.org/anthology/W16-2811/.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 2227–2237. ACL, 2018.

David Pizarro. Nothing more than feelings? the role of emotions in moral judgment. *Journal for the Theory of Social Behaviour*, 30(4):355–375, 2000.

Martin Potthast, Lukas Gienapp, Florian Euchner, Nick Heilenkötter, Nico Weidmann, Henning Wachsmuth, Benno Stein, and Matthias Hagen. Argument search: Assessing argument relevance. In *Proceedings of the 42nd SIGIR*, pages 1117–1120, 2019.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. Contextual stance classification of opinions: A step towards enthymeme reconstruction in online reviews. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 31–39. Association for Computational Linguistics, 2016. doi:10.18653/v1/W16-2804.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL `http://arxiv.org/abs/1908.10084`.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. Classification and Clustering of Arguments with Contextualized Word Embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pages 567–578. Association for Computational Linguistics, 2019.

Daniel E Rose and Danny Levinson. Understanding user goals in web search. In *Proceedings of the 13th WWW*, pages 13–19, 2004.

Hojjat Salehinejad, Sharan Sankar, Joseph Barfett, Errol Colak, and Shahrokh Valaee. Recent advances in recurrent neural networks. *arXiv preprint arXiv:1801.01078*, 2017.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

Misa Sato, Kohsuke Yanai, Toshinori Miyoshi, Toshihiko Yanase, Makoto Iwayama, Qinghua Sun, and Yoshiki Niwa. End-to-end argument generation system in debating. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 109–114, 2015.

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. Aspect-controlled neural argument generation. *arXiv preprint arXiv:2005.00084*, 2020.

Shalom H Schwartz. Are there universal aspects in the structure and contents of human values? *Journal of social issues*, 50(4):19–45, 1994.

Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. Societal biases in language generation: Progress and challenges. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4275–4293, 2021.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, 2021.

Kenneth Skiba, Matthias Thimm, Andrea Cohen, Sebastian Gottifredi, and Alejandro J García. Abstract argumentation frameworks with fallible evidence. In *Computational Models of Argument*, pages 347–354. IOS Press, 2020.

Noam Slonim, Gurinder Singh Atwal, Gašper Tkačik, and William Bialek. Information-based clustering. *Proceedings of the National Academy of Sciences*, 102(51):18297–18302, 2005.

Noam Slonim, Yonatan Bilu, Carlos Alzate, Roy Bar-Haim, Ben Bogin, Francesca Bonin, Leshem Choshen, Edo Cohen, Lena Dankin, Lilach Edelstein, Liat Ein Dor, Roni Friedman-Melamed, Assaf Gavron, Ariel Gera, Martin Gleize, Shai Gretz, Dan Gutfreund, Alon Halfon, Daniel Hershcovich, and Ranit Aharonov. An autonomous debating system. *Nature*, 591:379–384, 03 2021. doi:10.1038/s41586-021-03215-w.

Samuel Sousa and Roman Kern. How to keep text private? a systematic review of deep learning methods for privacy-preserving natural language processing. *Artificial Intelligence Review*, 56(2):1427–1492, 2023.

Maximilian Spliethöver and Henning Wachsmuth. Argument from old man's view: Assessing social bias in argumentation. In *Proceedings of the 7th Workshop on Argument Mining*, pages 76–87, Online, December 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.argmining-1.9`.

Christian Stab and Iryna Gurevych. Annotating Argument Components and Relations in Persuasive Essays. In *Proceedings of the the 25th International Confer-*

ence on Computational Linguistics, Dublin, Ireland, 2014a. Dublin City University and Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. Identifying Argumentative Discourse Structures in Persuasive Essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 46–56. Association for Computational Linguistics, 2014b.

Christian Stab and Iryna Gurevych. Recognizing insufficiently supported arguments in argumentative essays. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 980–990, 2017.

Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. Cross-topic Argument Mining from Heterogeneous Sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*, pages 3664–3674, Brussels, Belgium, November 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D18-1402.

Manfred Stede and Jodi Schneider. Argumentation mining. *Synthesis Lectures on Human Language Technologies*, 11(2):1–191, 2018.

Shahbaz Syed, Khalid Al Khatib, Milad Alshomary, Henning Wachsmuth, and Martin Potthast. Generating informative conclusions for argumentative texts. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3482–3493, 2021.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-Based Methods for Sentiment Analysis. *Computational Linguistics*, 37 (2):267–307, 06 2011. ISSN 0891-2017. doi:10.1162/COLI_a_00049. URL https://doi.org/10.1162/COLI_a_00049.

Zhixing Tan, Shuo Wang, Zonghan Yang, Gang Chen, Xuancheng Huang, Maosong Sun, and Yang Liu. Neural machine translation: A review of methods, resources, and tools. *AI Open*, 1:5–21, 2020. ISSN 2666-6510. doi:https://doi.org/10.1016/j.aiopen.2020.11.001. URL https://www.sciencedirect.com/science/article/pii/S2666651020300024.

Stephen Toulmin. *The Uses of Argument*. Cambridge University Press, 1958.

France van Eemeren and Peter Houtlosser. Strategic Manoeuvring in Argumentative Discourse. *Discourse Studies*, 1(4):479–497, 1999. doi:10.1177/1461445699001004005.

Frans H Van Eemeren and Rob Grootendorst. *A systematic theory of argumentation: The pragma-dialectical approach*. Cambridge University Press, 2004.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Henning Wachsmuth and Till Werner. Intrinsic Quality Assessment of Arguments. In *28th International Conference on Computational Linguistics (COLING 2020)*, pages 6739–6745. International Committee on Computational Linguistics, December 2020. doi:10.18653/v1/2020.coling-main.592. URL https://aclanthology.org/2020.coling-main.592.

Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Alberdingk Tim Thijm, Graeme Hirst, and Benno Stein. 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, 2017a. URL http://aclweb.org/anthology/E17-1017.

Henning Wachsmuth, Martin Potthast, Khalid Al-Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein. Building an Argument Search Engine for the Web. In Kevin Ashley, Claire Cardie, Nancy Green, Iryna Gurevych, Ivan Habernal, Diane Litman, Georgios Petasis, Chris Reed, Noam Slonim, and Vern Walker, editors, *4th Workshop on Argument Mining (ArgMining 2017) at EMNLP*, pages 49–59. Association for Computational Linguistics, September 2017b. URL https://www.aclweb.org/anthology/W17-5106.

Henning Wachsmuth, Benno Stein, and Yamen Ajjour. "PageRank" for Argument Relevance. In Phil Blunsom, Alexander Koller, and Mirella Lapata, editors, *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2017)*, pages 1116–1126. Association for Computational Linguistics, April 2017c. URL http://aclweb.org/anthology/E17-1105.

Henning Wachsmuth, Manfred Stede, Roxanne El Baff, Khalid Al Khatib, Maria Skeppstedt, and Benno Stein. Argumentation synthesis following rhetorical strategies. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3753–3765. Association for Computational Linguistics, 2018a. URL http://aclweb.org/anthology/C18-1318.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 241–251. Association for Computational Linguistics, 2018b. URL http://aclweb.org/anthology/P18-1023.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, 2020.

Douglas Walton. Objections, rebuttals and refutations. 2009.

Douglas Walton, Christopher Reed, and Fabrizio Macagno. *Argumentation Schemes*. Cambridge University Press, 2008.

Dingding Wang, Shenghuo Zhu, Tao Li, and Yihong Gong. Comparative document summarization via discriminative sentence selection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 7(1):1–18, 2013.

Lu Wang and Wang Ling. Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57, San Diego, California, June 2016. Association for Computational Linguistics. doi:10.18653/v1/N16-1007. URL https://www.aclweb.org/anthology/N16-1007.

Mayur Wankhade, Annavarapu Chandra Sekhara Rao, and Chaitanya Kulkarni. A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, 55(7):5731–5780, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Matti Wiegmann, Khalid Al Khatib, Vishal Khanna, and Benno Stein. Analyzing persuasion strategies of debaters on social media. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6897–6905, 2022.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*, 2019.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of*

the 2020 conference on empirical methods in natural language processing: system demonstrations, pages 38–45, 2020.

Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. A survey of knowledge-enhanced text generation. *ACM Computing Surveys (CSUR)*, 2022.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

Zhihan Zhang, Wenhao Yu, Mengxia Yu, Zhichun Guo, and Meng Jiang. A survey of multi-task learning in natural language processing: Regarding task relatedness and training methods. *arXiv preprint arXiv:2204.03508*, 2022.

Ingrid Zukerman, Richard McConachy, and Sarah George. Using argumentation strategies in automated argument generation. In *INLG'2000 Proceedings of the First International Conference on Natural Language Generation*, pages 55–62, Mitzpe Ramon, Israel, June 2000. Association for Computational Linguistics. doi:10.3115/1118253.1118262. URL `https://www.aclweb.org/anthology/W00-1408`.