*Computational Models of Argument*
*F. Toni et al. (Eds.)*


*doi:10.3233/FAIA220138*

# Generating Contrastive Snippets for Argument Search

Milad ALSHOMARY, Jonas RIESKAMP and Henning WACHSMUTH
*Paderborn University, Paderborn, Germany*

**Abstract.** In argument search, snippets provide an overview of the aspects discussed by the arguments retrieved for a queried controversial topic. Existing work has focused on generating snippets that are representative of an argument's content while remaining argumentative. In this work, we argue that the snippets should also be *contrastive*, that is, they should highlight the aspects that make an argument unique in the context of others. Thereby, aspect diversity is increased and redundancy is reduced. We present and compare two snippet generation approaches that jointly optimize representativeness and contrastiveness. According to our experiments, both approaches have advantages, and one is able to generate representative yet sufficiently contrastive snippets.

**Keywords.** snippet generation, argument search, argument presentation

## 1. Introduction

Most search engines present results along with short text excerpts of the underlying documents, so called *snippets*, in order to let users quickly assess the relevance of the results to their information needs [1]. In argument search, where the goal is to retrieve pro and con arguments on a queried controversial topic [2,3], snippets are of particular importance to provide an efficient overview of the spectrum of topical *aspects* covered by the retrieved arguments—without the need to go through all of them [4].

Standard snippet generation focuses on the overlap of the input query with the document [5,6], or abstractions thereof [7]. In the context of argument search, however, Alshomary et al. [4] argued in favor of snippets that summarize an argument's main claim and the main reason supporting the claim. In their experiments, the authors demonstrated that snippets generated towards this goal are more representative than generic content summaries and query-dependent snippets.

In this paper, we highlight the limitations of such argument snippet generation and propose an extended setting for the task in order to maximize the usability of the resulting snippets for argument search engines. Particularly, the extractive summarization approach of Alshomary et al. [4] addressed two goals of snippet generation: *representativeness* and *argumentativeness*. However, the top-ranked arguments retrieved by an argument search engine usually discuss the same queried controversial topic. Hence, an approach that aims to extract the main claim of an argument will tend to generate semantically similar snippets for several arguments. This behavior is highlighted in Figure 1, where two pro arguments are shown for the query "tuition fees". Here, a focus on

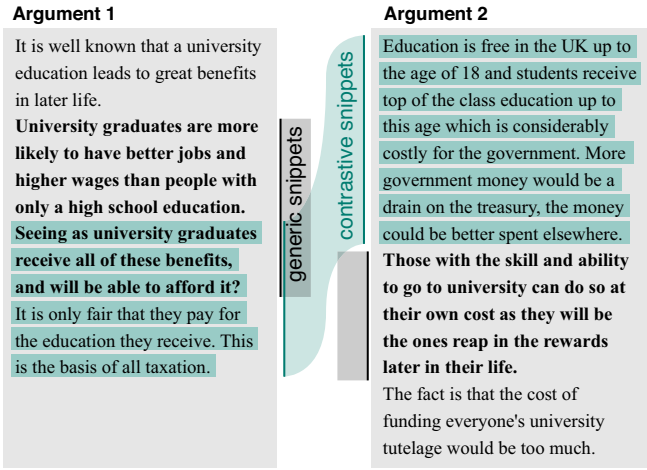


**Figure 1.** Example arguments returned for the query "tuition fees" (we show only two here for simple illustration). In each case, the bold sentences represent a generic snippet for the respective argument, whereas those with a colored background form a more contrastive argument snippet.

only representativeness and argumentativeness would likely produce similar snippets for both arguments (e.g., based on the sentences marked bold), reducing the argument search engine's capability to provide an effective aspect overview.

To alleviate the outlined limitation, we propose to additionally maximize the *contrastiveness* of a snippet. We define an argument snippet as contrastive, if it highlights the uniqueness of the input argument compared to other arguments from the same context (say, those from the same result page in argument search). The input to our approach is a set of arguments from the same context. The output is a set of snippets that are argumentative and representative of their argument while being contrastive toward the others. By focusing more on contrastiveness, the new snippets (shown with a colored background in Figure 1) increase the coverage of the diverse aspects discussed in the input arguments. Naturally, achieving higher contrastiveness might result in lower representativeness; a trade-off that can be adjusted, as we show later in experiments.

We approach the extended task setting in two ways. First, we extend the graph-based approach of Alshomary et al. [4], which ranks sentences based on their centrality and argumentativeness, by contrastiveness. Here, we encode an extra term to discount the sentence's similarity to other arguments. Second, we exploit the resemblance of our setting to the comparative summarization task [8]. Concretely, we adapt the approach of Bista et al. [9] who model the latter task as the selection of a snippet that a powerful classifier can distinguish from other arguments but not from the input argument.

In our experiments, we evaluate the approaches on a dataset constructed from the corpus of the argument search engine *args.me*. In particular, we use controversial topics from Wikipedia along with entries from the query log of args.me itself [10] to retrieve argument collections from the args.me API. We quantify representativeness by computing the cosine similarity of a snippet with the average embedding of its argument, and we measure argumentativeness as a quality dimension using the model of [11] trained to measure the argumentative quality of sentences. For contrastiveness, finally, we adapt silhouette analysis [12] as a proxy to measure the quality of clusters whose centroids are

the generated snippets. Our results demonstrate the trade-off between representativeness and contrastiveness, and they indicate how to balance this trade-off.

In a subsequent user study, we manually compared our approaches to a baseline that focuses only on representativeness. We demonstrate that both approaches generate snippets that highlight the arguments' uniqueness better, whereas the comparative summarization approach produces more representative snippets. Our findings support the applicability and importance of considering the contrastiveness of a snippet within argument search. For reproducibility, we make our code and resources publicly available.[1]

## 2. Related Work

For snippet generation in general, abstractive and extractive summarization techniques have been explored [13,14]. In both cases, a user's query may be considered during generation (query-dependent snippet) or disregarded (query-independent). Alshomary et al. [4] suggest that query-independent snippets are more suitable in an argument search scenario. Therefore, we also consider such snippets in this paper. The authors proposed a graph-based approach that forms snippets by selecting the two most important sentences in terms of their centrality in context and their argumentativeness. In contrast, we argue that a snippet should also highlight the argument's uniqueness in its context to maximize the diversity of snippets when presented to the end-user.

A branch of summarization research focuses on comparative summarization. Given a set of document groups, the task here is to generate summaries that are useful in comparing the differences between these groups [8,15]. Similarly, our goal is to obtain snippets that emphasize the differences between texts. However, our input is a set of arguments rather than groups of documents. Thus, we modify the recent comparative summarization approach of Bista et al. [16] by considering different surface features to maintain the snippet's argumentativeness.

There exists a body of research on the diversification of search results [17] that aims to retrieve diverse results while maintaining relevance to the queried topic in order to provide wider perspective. Differently, in this work, we are already given a set of arguments retrieved for some topic, and we aim to generate diverse snippets, where each snippet highlights the unique argumentative part of its argument.

## 3. Approaches

For the task of contrastive argument snippet generation, we define the input to be a set of $k \geq 2$ arguments $\mathbf{A} = \{A_1, \ldots, A_k\}$ from the same context, for example, all arguments from a search engine's result page. 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 one subset $S \subseteq A$ for each $A$, consisting of all sentences of the snippet (we predefine $|S| = 2$ in our experiments). Ideally, $S$ is representative of $A$, argumentative on its own, and contrastive towards all arguments in $\mathbf{A} \setminus \{A\}$.

In this section, we propose two alternative approaches to the defined task. The first, *Contra-PageRank*, extends the work of Alshomary et al. [4] by modeling the dissimilarity

[1] https://github.com/MiladAlshomary/contrastive-snippet-generation

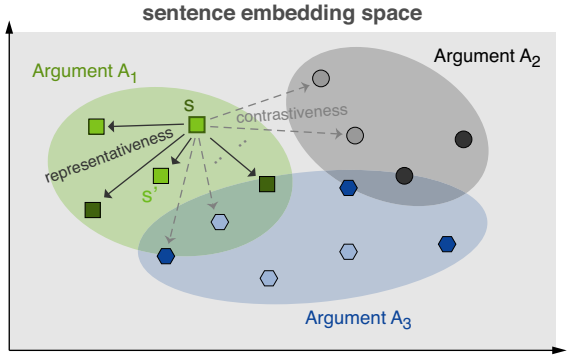


**Figure 2.** The idea of both approaches, illustrated for three arguments in a sentence embedding space. A sentence $s$ is used for a snippet of an argument $A_1$, if its joint representativeness and contrastiveness are higher than for other sentences $s'$ of the same argument. Argumentativeness (brighter symbols) is considered as well.

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. [16] to select a snippet $S$ that can be distinguished from $\mathbf{A} \setminus \{A\}$ but not from $A$. Both thus follow the idea to include a sentence $s$ in $S$, if it is both representative of $A$ and contrastive to $\mathbf{A} \setminus \{A\}$. We illustrate this idea in Figure 2.

### *3.1. Graph-based Summarization*

Alshomary et al. [4] proposed a graph-based approach that utilizes PageRank [18] to score sentences in terms of their centrality in context and argumentativeness. The two top-scored sentences are then extracted in their original order to form a snippet. 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 $A$ 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:

$$P(s_i) := \quad d_1 \cdot \sum_{s_j \in A, i \neq j} \frac{sim(s_i, s_j)}{\sum_{s_k \in A, j \neq k} sim(s_j, s_k)} \cdot P(s_j) \tag{1}$$

$$+ d_2 \cdot \frac{arg(s_i)}{\sum_{s_j \in A} arg(s_j)} \quad - \quad d_3 \cdot \frac{sim(s_i, \mathbf{A} \setminus \{A\})}{\sum_{s_j \in A} sim(s_j, \mathbf{A} \setminus \{A\})} \tag{2}$$

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 the original approach [4], 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 $sim(s_i, s_j)$. Finally, PageRank is applied to generate a score $P(s)$ for each $s$. As in [4], we start with equal initial scores for all sentences and iteratively update them until near-convergence. We rank all sentences of a given argument $A$ by score and generate a snippet from the two top-ranked sentences concatenated in their original order.

### 3.2. Comparative Summarization

Given the resemblance of our task to comparative summarization, we model the task in line with the mentioned approach of Bista et al. [16]: For an argument $A$, the goal is to find snippet sentences $S \subseteq A$ subject to (a) $S$ being representative of $A$, and (b) $S$ being contrastive to $\mathbf{A} \setminus \{A\}$. This 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 the snippet's 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 the snippet's 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, [9] used maximum mean discrepancy (MMD) [19] as an estimation of the classifiers' effectiveness. Given a set of arguments $\mathbf{A}$, the goal is then to find the snippet sentences $S \subseteq A$ of all arguments $A \in \mathbf{A}$ that maximize the following term:

$$\sum_{A \in \mathbf{A}} \left( -MDD^2(S, \{A\}) + \lambda \cdot MMD^2(S, \mathbf{A} \setminus \{A\}) \right) \tag{3}$$

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

However, there may be other features that signal sentence importance which are not reflected by similarity (e.g., argumentativeness in our case). For this, [16] 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 $\mathscr{T}$ with tuples of argument $A$, generic snippet $\bar{S} \subseteq A$, and context $\mathbf{A}$, the goal is to minimize (note the switched signs) the following adjusted term:

$$\frac{1}{|\mathscr{T}|} \sum_{(A, \bar{S}, \mathbf{A}) \in \mathscr{T}} \left( MDD^2(\bar{S}, A, \theta) - \lambda \cdot MMD^2(\bar{S}, \mathbf{A} \setminus \{A\}, \theta) \right) \tag{4}$$

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 [20]
6. *Argumentativeness.* Count of words from a claim lexicon [4]

## 4. Experiments

We now analyze the trade-off between representativeness and contrastiveness in snippet generation, and we explore how to adjust it via hyper-parameters of the two approaches. We will present the data collection and preprocessing, implementation details, as well as the automatic and manual evaluations that we carried out.

### *4.1. Data*

For evaluation, we need a dataset of arguments grouped into contexts. Since our work is motivated by the idea of argument search engines, we use the *args.me* corpus of [10] as the source. Particularly, 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 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.[4]

### *4.2. Implementation Details*

For both approaches, we preprocess all input arguments in a number of cleansing steps, namely we remove debate artifacts[5], references, enumeration symbols, and sentences shorter than three characters. We measured sentences similarity in terms of the cosine of their embeddings generated with Sentence-BERT [22]. In the following, we give further implementation details of the two approaches.

*Contra-PageRank* 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.

*Comp-Summarizer* To obtain ground-truth generic snippets $\bar{S}$ that are necessary for the supervised training, we followed the previous work [4] in considering the argument's conclusion a proper generic snippet. To this end, we used the *args.me* corpus and heuristically generated generic snippets based on the sentences' overlap with the conclusion.[6] We assessed different combinations of values for the hyperparameters, including the contrastiveness weight $\lambda$. 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

[4] All three collected datasets will be made publicly available upon acceptance.

[5] The artifacts are mostly utterances of social interaction between debaters [21].

[6] The algorithm used is the one that was proposed by Bista et al. [16].

**Table 1.** Automatic evaluation scores of *Contra-PageRank* for three selected combinations of hyperparameter values. The best value in each column is marked in bold.

| $d_1$ | $d_2$ | $d_3$ | **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 |

### *4.3. Automatic Evaluation*

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 $\mathbf{S} = \{S_1, \ldots S_k\}$ generated for a set of arguments $\mathbf{A} = \{A_1, \ldots A_k\}$, we pseudo-cluster the embedding of all arguments' sentences, with each snippet $S_i$ as one centroid.[7] 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 [11] trained on a regression task to predict the argumentative quality score of a sentence.[8]

*Results* Table 1 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.[9]

In Table 2, we explore the trade-off between representativeness and argumentativeness for *Comp-Summarizer*, showing evaluation scores for selected values of the contrastiveness weight $\lambda$. Analogously, a higher $\lambda$ 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.

### *4.4. 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, none of which was an author of this paper (university students with good English

[7] A snippet's embedding is averaged from its sentences' embeddings.
[8] We implemented the topic-independent version of the model.
[9] The effect of adding argumentativeness was also rather low in the original paper [4].

**Table 2.** Automatic evaluation scores of *Comp-Summarizer* for three different values of the contrastiveness weight $\lambda$. The best value in each column is marked in bold.

| $\lambda$ | 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.** 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.

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

skills). We chose the variants of the two approaches that yielded best contrastiveness above: the third row of Table 1 for *Contra-PageRank*, and the third of Table 2 for *Comp-Summarizer*. As a baseline focusing on representativeness, we also included the *Contra-PageRank* variant in the first row of Table 1, which is similar to the approach of [4], except for computing centrality only based on the argument's sentences rather the whole context. Accordingly, we refer to this 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.[10] 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 [23].

*Results* The average inter-annotator agreement of the two annotator pairs was substantial, 0.74 in terms Krippendorff's $\alpha$, suggesting reliable results. Table 3 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

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

**Table 4.** Example arguments on *Cloud Seeding* along with the snippet generated for each by our two approaches and the baseline

| **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.<br><br>**- ArgPageRank's Snippet:** Cloud seeding should be used in certain areas<br>**- Comp-Summarizer's snippet:** Cloud seeding is safe and harmless to the environment according to both EcoHearth.com and Weather Modifications.org<br>**- 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.<br><br>**- 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<br>**- 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<br>**- 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.<br><br>**- ArgPageRank's Snippet:** There will be no new evidence or arguments to be formed during this round.<br>**- 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.<br>**- Contra-PageRank's snippet:** Cloud seeding should not be used because the chemical is still dangerous and can cause serious harm to the environment. |

achieved by *Comp-Summarizer* which generated the most representative snippets while maintaining contrastiveness almost as often as *Contra-PageRank* (81%).

*Example analysis* In Table 4, 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 ArgPageRank tends to select general sentences like *"Cloud seeding should be used in certain areas."* or *"no new evidence or arguments to be found.."*, while CompSummarizer generated snippets that focus on aspects unique to the argument like *scientific consensus"* and *"harmless to the environment"*.

## 5. Conclusion

In this work, we argued for the importance of *contrastive* snippets in argument search, that is, snippets that emphasize an argument's unique aspects in the context of others. Building on related work, we have proposed two extractive summarization approaches. Despite room for improvement, our experiments showed their effectiveness and the inhererent trade-off between snippet's contrastiveness and representativeness. While the graph-based summarizer turned out to foster contrastiveness most, the comparative summarizer seems to balance the trade-off better. By focusing on both representativeness and representativeness, we believe that argument snippet generation can produce snippets that help in distinguishing different arguments efficiently.

## 6. Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation): TRR 318/1 2021 - 438445824. We would also like to thank the reviewers and the participants who took part in the manual evaluation of our methods.

## References

[1] Marcos MC, Gavin F, Arapakis I. Effect of Snippets on User Experience in Web Search. In: Proceedings of the 16th HCI; 2015. p. 47.

[2] Wachsmuth H, Potthast M, Al-Khatib K, Ajjour Y, Puschmann J, Qu J, et al. Building an Argument Search Engine for the Web. In: Proceedings of the 4th Workshop on Argument Mining; 2017. p. 49-59.

[3] Daxenberger J, Schiller B, Stahlhut C, Kaiser E, Gurevych I. Argumentext: argument classification and clustering in a generalized search scenario. Datenbank-Spektrum. 2020;20(2):115-21.

[4] Alshomary M, Düsterhus N, Wachsmuth H. Extractive snippet generation for arguments. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval; 2020. p. 1969-72.

[5] White RW, Ruthven I, Jose JM. Finding relevant documents using top ranking sentences: an evaluation of two alternative schemes. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval; 2002. p. 57-64.

[6] Penin T, Wang H, Tran T, Yu Y. Snippet generation for semantic web search engines. In: Asian Semantic Web Conference. Springer; 2008. p. 493-507.

[7] Chen WF, Syed S, Stein B, Hagen M, Potthast M. Abstractive Snippet Generation. In: Proceedings of The Web Conference 2020; 2020. p. 1309-19.

[8] Wang D, Zhu S, Li T, Gong Y. Comparative document summarization via discriminative sentence selection. ACM Transactions on Knowledge Discovery from Data (TKDD). 2013;7(1):1-18.

[9] Bista U, Mathews A, Shin M, Menon AK, Xie L. Comparative document summarisation via classification. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33; 2019. p. 20-8.

[10] Ajjour Y, Wachsmuth H, Kiesel J, Potthast M, Hagen M, Stein B. Data Acquisition for Argument Search: The args. me Corpus. In: Proceedings of the KI; 2019. p. 48-59.

[11] Gretz S, Friedman R, Cohen-Karlik E, Toledo A, Lahav D, Aharonov R, et al. A large-scale dataset for argument quality ranking: Construction and analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34; 2020. p. 7805-13.

[12] Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics. 1987;20:53-65.

[13] Croft B, Metzler D, Strohman T. Search Engines: Information Retrieval in Practice. 1st ed. Addison-Wesley; 2009.

[14] Gholamrezazadeh S, Salehi MA, Gholamzadeh B. A Comprehensive Survey on Text Summarization Systems. In: Proceedings of the 2nd CSA; 2009. p. 1-6.

[15] Li L, Zhou K, Xue GR, Zha H, Yu Y. Enhancing diversity, coverage and balance for summarization through structure learning. In: Proceedings of the 18th international conference on World wide web; 2009. p. 71-80.

[16] Bista U, Mathews AP, Menon AK, Xie L. SupMMD: A Sentence Importance Model for Extractive Summarisation using Maximum Mean Discrepancy. In: Cohn T, He Y, Liu Y, editors. Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020. vol. EMNLP 2020 of Findings of ACL. Association for Computational Linguistics; 2020. p. 4108-22. Available from: https://doi.org/10.18653/v1/2020.findings-emnlp.367.

[17] Maxwell D, Azzopardi L, Moshfeghi Y. The impact of result diversification on search behaviour and performance. Information Retrieval Journal. 2019;22(5):422-46.

[18] Page L, Brin S, Motwani R, Winograd T. The PageRank Citation Ranking: Bringing Order to the Web. Stanford InfoLab; 1999.

[19] Gretton A, Borgwardt KM, Rasch MJ, Schölkopf B, Smola A. A Kernel Two-Sample Test. Journal of Machine Learning Research. 2012;13(1):723-73.

[20] Erkan G, Radev DR. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research. 2004;22:457-79.

[21] Dorsch J, Wachsmuth H. Semi-Supervised Cleansing of Web Argument Corpora. In: Proceedings of the 7th Workshop on Argument Mining. Online: Association for Computational Linguistics; 2020. p. 19-29. Available from: https://aclanthology.org/2020.argmining-1.3.

[22] Reimers N, Gurevych I. 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; 2019. Available from: https://arxiv.org/abs/1908.10084.

[23] Zhang J, Zhao Y, Saleh M, Liu P. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In: International Conference on Machine Learning. PMLR; 2020. p. 11328-39.