DRACS: Diachronic Representation of Argument Construction Styles

Yevhen Kostiuk, John Lawrence and Chris Reed[,*]

[a]Centre for Argument Technology (ARG-tech), University of Dundee, United Kingdom
ORCID (Yevhen Kostiuk): https://orcid.org/0000-0002-8104-5141, ORCID (John Lawrence): https://orcid.org/0000-0003-1162-097X, ORCID (Chris Reed): https://orcid.org/0000-0002-6849-1374

Abstract. In computational argument mining, the final constructed argument graph is usually the center of attention, with researchers analyzing its structure and developing methods to extract valuable insights from it. In this paper, we argue that analyzing only the final argument graph is not enough. We demonstrate that the same argument graphs do not imply the same argument construction styles (ways of populating them), which are an important abstract argumentation concept. Therefore, both final structure and the construction style should be considered when analyzing arguments.

To address the analysis of argument construction styles, we present the DRACS: **D**iachronic **R**epresentation of **A**rgument **C**onstruction **S**tyles framework to capture argument construction style features and enable their analysis alongside the graph structures. We evaluate our methodology on three distinct datasets of pre-defined similar and differing construction styles: the AIF-labeled corpora UNSC and US2016, as well as a custom corpus of synthetic tree-based argument graphs. We apply our method to highlight similarities and differences of argument construction styles on US2016 datasets. Our findings suggest that even if the final argument graphs display a high degree of similarity, their construction styles could differ significantly.

1 Introduction

A core concept of the argument mining field is the argument graph (AG) [18]. This graph represents how argument discourse units connect to each other and provides valuable insights on the structure, connectivity, potential strong and weak points of the argument etc. There are various results in the field that address a challenge of the argument graph generation [20, 12, 13]. The Argument Interchange Format (AIF) [11] is one way of presenting AG in a computationally friendly form.

However, when building an argument, people can use different *argument construction styles*. In the context of this work, *argument construction style* refers to the process by which an argument graph is constructed over time. Construction styles have a major impact on the different aspects of the argumentation: how easy it is to follow, how persuasive the argument is and so on. We use the word construction to indicate the process of building an argument over time during a discourse.

Argument construction styles do not emerge from the argument graph itself, they originate from human reasoning and language use. The way an argument progresses over time is influenced by linguistic choices, rhetorical techniques, and cognitive structuring of human reasoning, which makes them human-driven. This distinction aligns with research in abstract argumentation theory [9, 10], which emphasizes that argument structure and argument construction style are connected yet separate concepts.

Different argument styles do not necessarily imply different resulting argument graphs. Consider the example shown in Figure 1. This example demonstrates usage of two distinct construction styles that result in the same AG. The first style (upper row) presents support for the conclusion in a "breadth-first" way, while the second one (bottom row) in a "depth-first" way. By analyzing only the structure of the final argument graph, it is impossible to retrieve this difference, which could lead to the incorrect conclusion of them being the same.

In this paper, we address the challenge of the automatic analysis of argument construction styles. We introduce the DRACS: **D**iachronic **R**epresentation of **A**rgument **C**onstruction **S**tyles framework that represents the argument construction styles as multidimensional times series. Our framework works entirely on the structure level and does not use semantics of the arguments. Our methodology is interpretable and can be easily modified to the custom case.

Our contributions are:

- Development of an interpretable computational method to represent the argument construction style as the multidimensional vector time series. We evaluated our framework on the publicly available AIF datasets by testing framework's ability to distinguish between similar and dissimilar argument styles with the same AG.
- Application of our framework on the labeled US 2016 Republican and Democrats presidential primaries and analyzed candidates' argument styles from different perspectives: pair-wise and clustering differences. Our findings suggest that some candidates have distinct argument construction styles even when sometimes having very similar argument graphs.

The code can be found in the project's GitHub repository[1].

2 Related Work

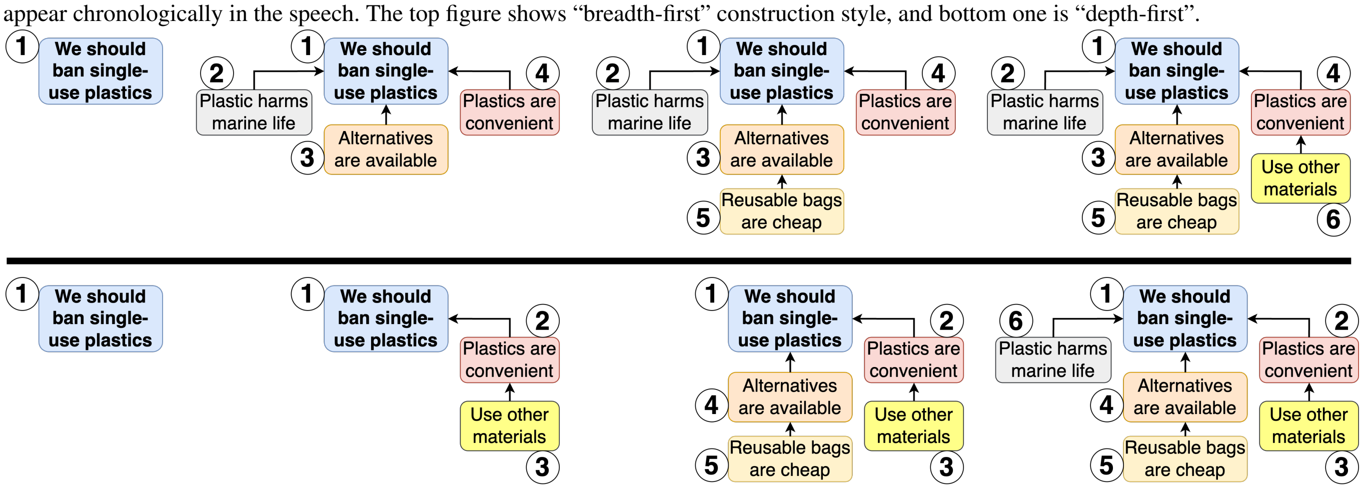
At the time of writing, in the argument mining domain there were no available frameworks similar to proposed one.

In [1], the authors analyzed the usage of argument "strategies" among different topics in argumentative editorials from the New

* Corresponding Author. Email: c.a.reed@dundee.ac.uk.

[1] https://github.com/arg-tech/dracs

Figure 1: Example of the same argument graph, but different argument construction styles. The circles indicate the order of argument units that appear chronologically in the speech. The top figure shows "breadth-first" construction style, and bottom one is "depth-first".



York Times dataset. Their findings indicate that such argument strategies are associated with topic, but their analysis excludes temporal aspects of strategies that are the focus of argument construction style (ACS). We defined ACS as the process of populating an argument graph with nodes and connections over time.

Abstract dialectical frameworks (ADF) [9, 10] can be considered as generalizations of the classical Dung-style argumentation frameworks. ADFs provide toolkits for the argument analysis based on the acceptance of the arguments. In [3], timed ADF was introduced. In [6], the Timed Abstract Argumentation Frameworks (TAFs) were proposed. TAFs introduce temporal availability: arguments exist or are available only during specific time intervals with attack relations being time-dependent (they're only effective when both the attacker and target are available simultaneously). TAFs describe how argument acceptability changes over time due to availability and semantic load, while our approach captures patterns and differences in how arguments are constructed over time without semantic features. In [7], the authors explored the timed concurrent language for argumentation (TCLA) that allow interactions between communicating agents through argumentation processes, also taking into account the temporal duration of the performed actions. The authors provided functional implementation of the TCLA syntax with the user interface. In [8], the methodology was introduced to analyze temporal acceptability of the arguments. The authors provided the concept of defense, utilizing a set of criterion functions to measure the significance of the attack with the probability over time. However, all of these works rely on the Dung-style argumentation theory, require additional definition of acceptance function and objective of the argument, and consider the temporal implication of the arguments, not the way the argument is built. In our work, we introduce a flexible way to represent a process of the argument construction without additional objective or acceptance function in the vector space.

The representation of the arguments with respect to time was investigated (e.g. paragraph-level RNN-based argument representation for argument mining [23], dialogue vector models [21] etc.), yet the time-aware representations were not fully explored. A lot of research was conducted in the essay grading and teaching domain from the argument strategy (AS) view [17, 14, 22]. According to [27], the argument strategy can be considered as a composition of a set of assumptions, claims, and various types of evidences, utilizing argumentative language and structure. In [24], AS was defined as the set of rhetorical and stylistic tools that are employed to create an effective text. In the paper, the authors analyze an essay corpora from the persuasion standpoint. Author analyzed argument patterns in the graph and the semantic flow of the essay. Similarly, in [25], the method for identifying argumentative discourse structures in persuasive essays was introduced. The proposed method included two-step process: multiclass classification of the argument components and pair-wise relation classification of the components. After these steps, the set of structural, syntactic, contextual, and indicators features was extracted and the essay score was predicted. In this case, the chronology of the AG population is not considered fully, even though some of the features include location component.

The authors of [2] explored the differences of the argument strategies and linguistic realizations of the discussion sections of the Indonesian authors in the journals of Language Writing. The authors defined 5 different strategies that the writers were using. Their findings showed that there difference in the argument strategies between the sections in the local, national, and international journals.

The analysis and clustering of argument graphs were also conducted in the field: for feature selection [15], similarity-based retrieval of AIF argument graphs [4].

3 DRACS Framework

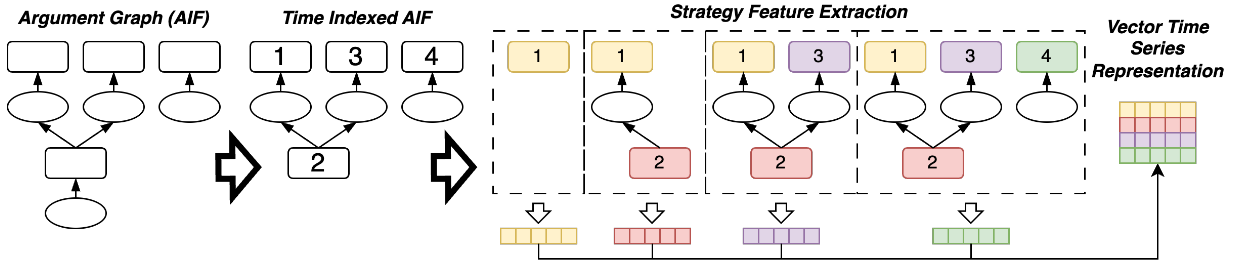
An overview of the DRACS framework is presented in Figure 2.

The framework represents the argument construction style (ACS) as multidimensional time series. Recall that ACS refers to the process of populating argument graph with nodes and connections over time. DRACS works on the structure level and does not consider semantics of the argument. In this work, we experimented with argument graphs in AIF format, but the framework is not limited to them and can be used with any other graph representation.

More formally, consider argument graph $G = (V, E)$ in AIF format, where V = I $\cup$ YA $\cup$ CA $\cup$ MA $\cup$ RA, I is a set of information nodes (I-nodes) expressing propositional content, YA a set of YA-nodes describing illocutionary intent, and the remainder standard AIF representations of argument relations: CA a set of CA-nodes capturing conflict relations, MA a set of MA-nodes capturing rephrase relations, and RA a set of RA-nodes capturing inference relations. $E = V \times V$ is a set of edges. Consider $i_j \in I, j = 1, \dots, N$.

Let $t : I \to \{1, \dots, N\}$, $t(i_j) = k$. For every l, m: $t(i_m) = t(i_l) \Leftrightarrow m = l$. The function $t(\cdot)$ maps each I-node with the *unique* time index k on when the content of the node was mentioned in the

Figure 2: DRACS pipeline overview. Firstly, the nodes in the AIF graph are mapped with the chronological time indices. After that, the features are extracted with the argument graph in the particular time index, creating a DRACS representation.



argument.

Consider $G_k = (V_k \subset V, E_k \subset E)$. $V_k = I_k \cup B_k \cup Y_k$, where $I_k = \{i_j | t(i_j) \leq k\}$, $B_k = \{v \in (V \setminus \text{YA}) \mid (i, v), (v, j) \in E; i, j \in I_k\}$, $Y_k = \{y \in \text{YA} \mid (y, b) \in E, b \in B_k\}$, and $E_k = \{(v_1, v_2) \mid v_1 \in V_k, v_2 \in V_k\}$. Finally, the vector of features f_k for the graph G_k is calculated, resulting in the vector time-series representation of the argument graph G.

As a feature set f_k we measure the following features: density, average degrees of the nodes (inner, outer), maximum inner and outer degree of the node, last added I-node inner and outer degree, longest path in the graph divided by the total number of nodes, average core number, number of leafs, number of weakly connected components, average number of nodes in the weakly connected components, and number of isolated I-nodes. This list of features is flexible and can be adapted for each use-case. To calculate the features, we used the networkX [16] Python package. Explanation of each feature is available via the documentation[2].

4 Data

We experimented on publicly available AIF-labeled datasets from AIFdb [19]: US 2016 primaries and presidential debates [28](US2016R1tv[3], US2016D1tv[4], and US2016G1tv[5]), UNSC dataset [30][6]. Additionally, we synthetically generated a dataset of tree-based AIF argument graphs (AGs) to increase the amount of data. All the datasets are available in the project's GitHub repository.

In each of the non-synthetic datasets we filtered argument graphs based on the amount of speakers. For the US2016R1tv, US2016D1tv, and US2016G1tv datasets, we selected only graphs that included only a single speaker (not counting moderators or hosts of the event). We ended up with argument graphs that contain a question or remark from the moderator and the candidate's speech as a response to it. For the UNSC dataset, we again selected the graphs that contain only one speaker. This dataset consists of speeches of members and guests of the UN Security Council. We excluded Russia from our analysis. For the remaining AGs, we removed locution information from it (L-nodes, TA-nodes, and corresponding YA-nodes).

To generate a synthetic dataset of tree-based AIF AGs, we used a two-step approach. First, we created the "depth" part by generating 4 to 10 I-nodes, each with a unique ID and a time index in ascending order. We then added a random relation node (RA, MA, or CA) between consecutive I-nodes. Next, each I-node had a 35% chance of having 3 to 6 branches. Each branch formed a new "depth" part with 2 to 5 connected I-nodes. Additionally, 2 to 5 random leaf nodes were added per branch. Finally, we connected a YA-node of random type to every node in the graph.

The statistics are presented in the Table 1. We obtained 150 samples for synthetic tree-based data, 196 for UNSC, 40 for US2016D, 5 for US2016G, 20 for US2016R. In total, we obtained 411 argument graphs.

Table 1: Counts statistics of different nodes per dataset and overall. **G** corresponds to US2016G, **R** corresponds to US2016R, and **D** corresponds to US2016D. **S.Tree** indicates a synthetic dataset.

		S.Tree	UNSC	D	G	R	Total
YA	***Avg***	348.09	28.21	55.58	30.4	25.35	147.5
	Std	110.14	13.28	37.71	21.13	9.84	166.97
I	***Avg***	174.55	13.71	36.1	18.6	16.25	74.77
	Std	55.07	4.62	23.95	11.41	5.14	83.33
RA	***Avg***	57.2	5.33	14.1	8.0	7.15	25.42
	Std	18.9	2.46	11.29	7.78	3.87	27.28
MA	***Avg***	58.61	3.6	4.2	4.0	1.43	26.46
	Std	18.95	2.49	3.08	1.0	0.53	29.82
CA	***Avg***	57.73	1.14	3.6	1.0	2.78	43.21
	Std	20.0	0.36	2.03	1.0	1.56	29.92
Edges	***Avg***	695.19	40.45	93.4	44.08	44.75	284.93
	Std	220.29	15.89	64.35	16.0	22.01	339.68

5 Evaluation

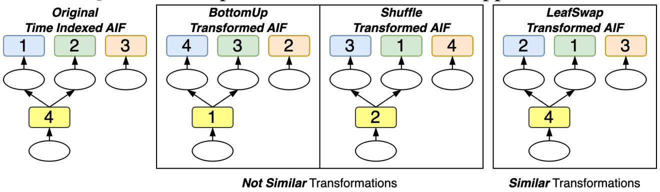
To evaluate the proposed framework, we tested its ability to distinguish between similar and dissimilar argument construction styles (ACS). An effective ACS representation should assign similar vectors to the similar ACS and different vectors for the dissimilar ones, based on a defined distance metric. As our distance score, we used Dynamic Time Warping (DTW) [5], which compares two vector time series. DTW is an algorithm that aligns two time-dependent sequences by warping their time axes to minimize the overall distance between corresponding points. It builds a cost matrix to find the optimal path that best matches the sequences, even if they differ in speed or length. The distance between similar ACS should be small, when distances between dissimilar ones expected to be large.

We defined the following transformations for this task: BottomUp, Shuffle, and LeafSwap for S. Each transformation takes as an input an argument graph a_1 and transforms it into the argument graph a_2. Both a_1 and a_2 share the same graph structure but differ in the time indices of their I-nodes. LeafSwap transformation was designed to make the ACS significantly different, but keeping the graph the same. BottomUp and Shuffle transformations were designed to make the ACS significantly different, keeping the same graph.

More specifically, the BottomUp transformation reverses the time occurrences of the original AIF, making nodes that were added last appear first, and vice versa. So, the nodes that were added at the end of the argument becoming the first ones and other way around. The Shuffle transformation randomly reassigns time occurrences to

[2] https://networkx.org/documentation/stable/reference/functions.html
[3] http://corpora.aifdb.org/US2016R1tv
[4] http://corpora.aifdb.org/US2016D1tv
[5] http://corpora.aifdb.org/US2016G1tv
[6] http://corpora.aifdb.org/UNSCPilotCorpus and http://corpora.aifdb.org/UNSCwps

Figure 3: Example of the transformations application.



I-nodes, effectively disrupting any inherent structure. The LeafSwap transformation randomly swaps the time occurrences of leaf (nodes that do not have outgoing connections) I-nodes that share the same parent. This transformation is only applicable to graphs containing leaf nodes. Examples of each transformation are presented in Figure 3.

For each available argument graph (AG) from the datasets, three pairs were created: (original AG, BottomUp transformed AG), (original AG, Shuffle transformed AG), and (original AG, LeafSwap transformed AG). Then, for each pair, we calculated the DTW distance scores from tslearn [26]. The greater the distance - the less similar the construction styles. The results are presented in the next section.

6 Results

Table 2 presents the DTW distance scores for each transformation across different datasets. The results confirm that the proposed framework effectively differentiates between similar and dissimilar ACS, demonstrating its validity as an argument construction style vector representation technique.

The LeafSwap transformation consistently produces near-zero distances, indicating that ACS remain highly similar despite minor modifications to the time indices of leaf I-nodes. This validates that LeafSwap does not introduce significant structural differences in argument development. The most similar scores are obtained for the synthetic dataset, as the dataset constructed in the synthetic way with leafs always present and having strict order structure.

Table 2: The Dynamic Time Warping (DTW) similarity scores per transformation.

Dataset	Method	Mean	Std	Med	Num
S.Tree	BottomUp	1,749.63	824.81	1,659.43	150
	Shuffle	1,362.88	685.8	1,209.82	150
	LeafSwap	0.0	0.0	0.0	150
UNSC	BottomUp	14.64	8.95	12.09	195
	Shuffle	15.82	9.03	13.81	182
	LeafSwap	1.15	3.17	0.0	55
US 2016 D	BottomUp	22.77	19.69	12.68	12
	Shuffle	17.74	14.45	12.03	11
	LeafSwap	0.0	0.0	0.0	1
US 2016 G	BottomUp	16.91	15.66	14.17	4
	Shuffle	30.48	30.7	21.72	3
US 2016 R	BottomUp	25.52	21.9	16.53	20
	Shuffle	20.26	19.41	10.11	19
	LeafSwap	0.8	2.11	0.0	7
Overall	BottomUp	698.56	993.03	23.8	381
	Shuffle	569.82	795.36	25.03	365
	LeafSwap	0.32	1.72	0.0	213

In contrast, BottomUp and Shuffle transformations yield significantly higher distances, suggesting that these transformations substantially alter ACS. Their overall mean distance (BottomUp: 698.56, Shuffle: 569.82) have orders of magnitude higher than for LeafSwap (0.32), showing the framework's ability to separate ACS. From a structural standpoint, the order of leaf nodes should not influence the graph construction process, which is clear in the synthetic case. The real world data has its values are around zero, indicating a almost identical distances.

Note, that for the *US 2016 G* dataset we were not able to apply LeafSwap transformation as the its available argument graphs did not contain leaf nodes. However, its values for available transformations are similar to the ones received for UNSC and other US 2016 datasets.

Different datasets are affected by transformations differently. In tree-based argument graphs, changing time indices have a stronger effect on the argument construction style (higher distances), while in the political debates datasets strategies might be more flexible, leading to lower distances in some cases.

To sum up, the results demonstrate that the use of DRACS and DTW-based distance is a reasonable measurement for distinguishing between similar and dissimilar ACS, supporting its potential application.

7 Applications

In this section, we demonstrate a practical application of DRACS by analyzing the televised US 2016 primary and presidential debates of Democratic and Republican candidates based on their transcripts. Each dataset is a collection of candidates' speeches. Our goal is to evaluate the consistency and variability of utilized argument construction styles (ACS) by different candidates and to extract their distinct argument styles. Additionally, we provide an interpretation of the differences based on the extracted DRACS features.

To achieve this, we performed **pairwise** and **cluster-based** comparisons of ACS. Pairwise comparison offers deeper insights into how individuals resemble each other on average, while cluster-based analysis helps identify argument patterns of the individual speech .

The dataset consists of transcripts from three events: Democratic primaries held on 2016-12-21, Republican primaries held on 2016-11-01, and Republican primaries held on 2016-12-21.

Each primary followed a structured format: candidates were introduced by a moderator, sometimes given a question or topic, and then delivered their speeches. For each speech, the dataset contains an argument graph of the speech in AIF format, where each I-node corresponds to the argumentative discourse units of the candidate or a moderator of the debate. Each AIF graph is the candidate's uninterrupted speech on a specific topic. If the candidate spoke multiple times in a debate, each speech is represented as a separate AG.

More formally, let $D = \{s_i^c |\ c \in C,\ i = 1, \ldots, N_c;\}$ denote a primary of a particular day, s_i^p refer to an argument graph (AG) of the speech given by a candidate c, N_c is a total amount of given speeches by c in the primary of that day, and C is total set of candidates.

To measure argument similarity from different perspectives, we define two distance metrics. Let AIF-D$(\cdot, \cdot)$ be a spectral graph [29] distance between two AIF argument graphs. Spectral graph distance is a measure between two graphs based on the eigenvalues, derived from their Laplacian matrices. Let Style-D$(\cdot, \cdot)$ be a DTW distance between two speeches' DRACS representations. Let G_d be the AIF-D distance matrix between the speeches and T_d be the Style-D distance matrix for each pair of the speeches. For each primary, we calculated those distances matrices separately.

To make a pairwise comparison, we averaged distances from G_d and T_d per candidate, obtaining G and T matrices. Each pair of candidates c_1, c_2 then mapped onto a two-dimensional space with coordinates from G and T: $(g_{c_1,c_2},\ t_{c_1,c_2})$. The values are standardized for comparability. The scaled points can be interpreted as follows:

1. $g_{c_1,c_2} > 0$ and $t_{c_1,c_2} > 0$. Argument graph and ACS are structurally different for both candidates. Therefore, they argue differently and arrive at the distinctly structured arguments.
2. $g_{c_1,c_2} < 0$ and $t_{c_1,c_2} > 0$. Argument graphs are structurally similar, but the ACS are different. This indicates that the candidates are using different construction styles, but end up with the same argument structure.
3. $g_{c_1,c_2} > 0$ and $t_{c_1,c_2} < 0$. Argument graphs are structurally different, but the ACS are similar. This indicates that the candidates are using similar construction styles, but end up with the different argument structure.
4. $g_{c_1,c_2} < 0$ and $t_{c_1,c_2} < 0$.Argument graphs and the ACS are structurally similar for both candidates. It shows that the candidates are using similar construction styles, as well as end up with the same argument structure.

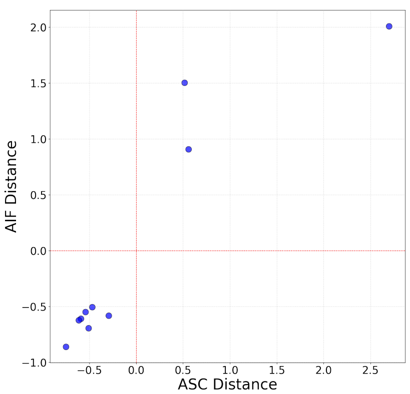
To perform cluster-based comparisons, we apply hierarchical clustering with average linkage separately to T_d and G_d. Unlike pairwise comparisons, clustering operates at the speech level and helps to understand argument construction styles of candidates per-topic and per-day.

7.1 US 2016 Democratic Party Presidential Primaries (TV) Analysis

Pair-wise comparison. Figure 4 presents a scaled plot of pairwise comparisons for each speaker.

In the figure, the Y-axis represents the scaled spectral argument graph distance, while the X-axis represents the scaled DTW distance of the DRACS representations ACS. The graphs are divided into quadrants for better visualization. Each point indicates a pair of speakers.

Figure 4: Pair-wise comparison of argument construction styles and argument graph distances of US 2016 D dataset (Democratic Party).

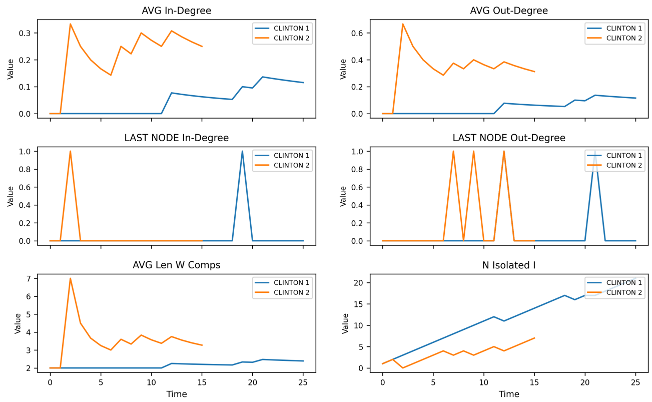


In the image, all the points located in the first or third quadrants. This suggests that the speakers either employ different ACS, leading to different final argumentation graphs, or use similar styles, resulting the similar argumentation graphs. Results indicate a high degree of consistency among the candidates.

Cluster analysis. Figure 6 presents clustering results as the dendrograms based on the different distance matrices.

The AIF-based clustering does not reveal clear separations among speakers. On the other hand, style-based clustering highlights a distinction: one of Hillary Clinton's speeches differs noticeably from the other. This difference arises from its connectivity: one speech lacked argument relations (as she was introducing herself, almost no argument connections were made), while the other exhibited more connections between nodes. The extracted features are presented in Figure 5.

Figure 5: Selected set of extracted DRACS representations of Clinton's speeches per-dimension. The Y-axis indicates a value of the particular feature that was calculated.



7.2 US 2016 Republican Party Presidential Primaries (TV) Analysis

Pair-wise comparison. Figure 7 presents a scaled plot of pairwise comparisons for each speaker.

Republicans demonstrate different patterns compared to Democrats. Some points appear in the fourth quadrant, indicating that some speakers employed similar styles but arrived at different final argumentation graphs. The speaker pairs from the fourth quadrant are highlighted on the Figure 8.

The recurring names in the fourth quadrant are Trump and Kasich.

Cluster analysis. Figure 9 presents clustering results as the dendrograms based on the different distance matrices on different days.

On 2016-11-01, the AIF-based clustering does not reveal any noticeable differences. However, style-based clustering suggests the presence of two or three distinct clusters, with Kasich standing out from other candidates. The investigation of his ACS time series per feature reveals that the Kasich's rolling degrees of the nodes are higher, the number of weakly connected components is lower (but its average length is higher), and the number of leafs is lower than in the other candidate's ACS. This finding suggests that Kasich's argument style is more interconnected (more edges), with fewer "topic jumps" and structural shifts (see Figure 10).

Two other groups can be described as follows: Walker and Huckabee (hereafter referred to as the WH group), and Paul, Cruz, Christie, and Trump. Based on feature graphs, the WH group exhibits a higher average node degree toward the end of their speeches, along with a distinct pattern in the evolution of weakly connected components (see 10)). This suggests that Walker and Huckabee progressively build their arguments, connecting more nodes as their speeches conclude. In contrast, the second group demonstrates a more chaotic structure, lacking clear similarities or trends.

On 2016-12-21, the AIF-based clustering reveals different patterns. Trump demonstrates a distinct argument structure, with two additional potential clusters. His ACS on this day shows higher node degrees towards the end of his speech, as well as the the longest path in the graph. He had a fewer amount of leafs and isolated nodes, and

Figure 6: Dendrograms of hierarchical clustering results for different distance matrices of US 2016 D dataset (Democratic Party).

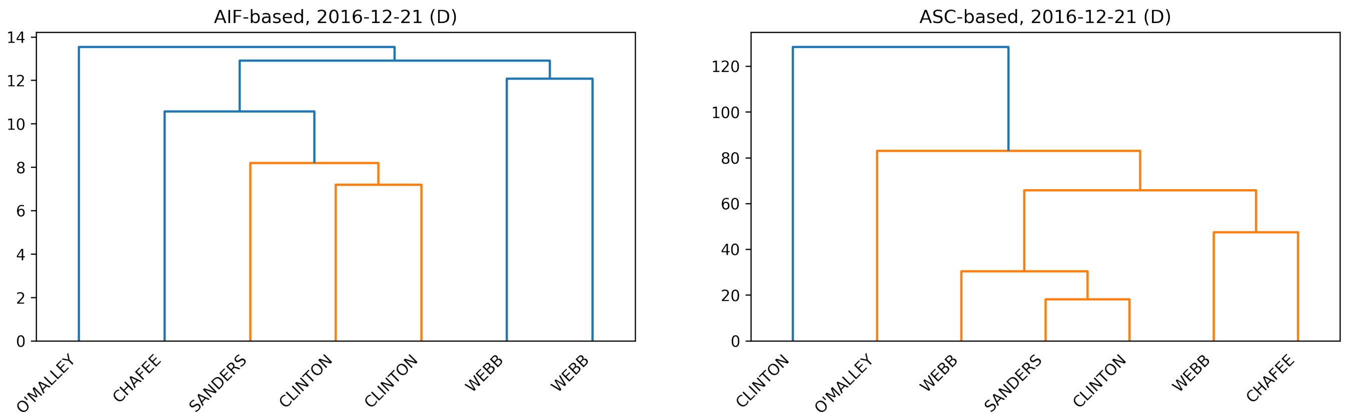
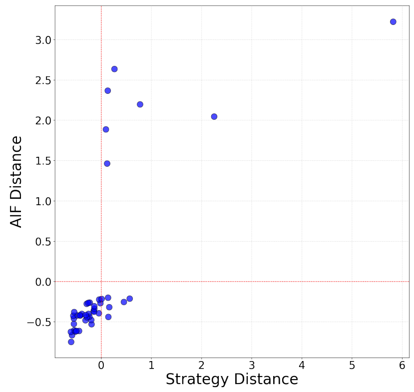


Figure 7: Pair-wise comparison plot of ACS and argument graph distances of US 2016 R dataset (Republican Party).



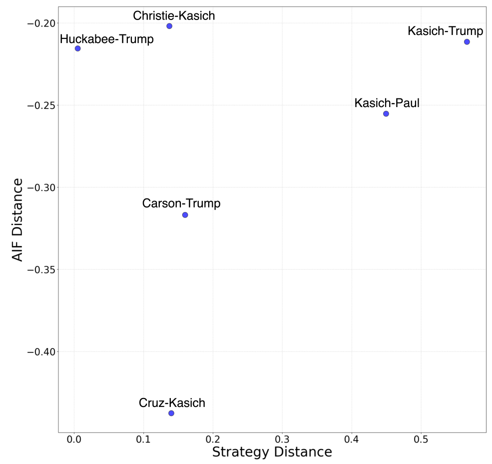


Figure 8: The fourth quadrant of pair-wise comparison of argument strategies and argument graph distances (Republican Debates).

only one weakly connected component, indicating better topic consistency throughout his speech.

Another noticeable AIF-based cluster is presented by Kasich. His differences are in a higher average core (centrality) score in the graph, large number of leafs, isolated nodes, and weakly connected components. High centrality shows that the candidate emphasizes key points in his speech.

The dataset for 2016-12-21 contained two speeches (introduction and closing) by Bush, which follow different patterns when it comes to ACS (see Figure 11). The closing speech (2 in the figure) has fewer discourse units in it than the introduction speech (1 in the figure).

Regarding style-based clustering, the analysis suggests 3–5 distinct clusters. One of the most prominent cluster is the Trump/Bush group, characterized by a higher degree values, a smaller number of leafs, isolated nodes, and components.

Finally, Paul's distinction can be caused by his speech being much shorter than others.

8 Conclusion

In this paper, we introduced DRACS, a framework for representing and analyzing argument construction style through time-series features. Our study highlights the importance of considering not only the final structure of argument graphs but also the construction styles by which they have been created. Through our analysis of the US2016 debates and other AIF-labeled corpora, we demonstrated that similar argument graphs can emerge from different argumentation construction styles.

Our findings emphasize that argument construction styles provide additional insights beyond structural analysis alone, highlighting individual differences and consistencies, as well as offering an approach to detect anomalies from a structural standpoint. Future work could extend our approach to real-time argument tracking, explore its applicability to other domains, and refine style-based clustering techniques.

The introduction of DRACS offers new insights for both computational argument mining and argumentation theory. While traditional approaches focus on the non-dynamic structure of argument graphs, DRACS emphasizes how these graphs are constructed over time. This temporal perspective introduces a novel feature space that can support tasks such as persuasion analysis, relation identification, authorship detection, style clustering etc. More broadly, the concept of argument construction style contributes a new dimension to discourse analysis by enriching both theoretical and applied perspec-

Figure 9: Dendrograms for different distance matrices for US 2016 R dataset (Republican Debates).

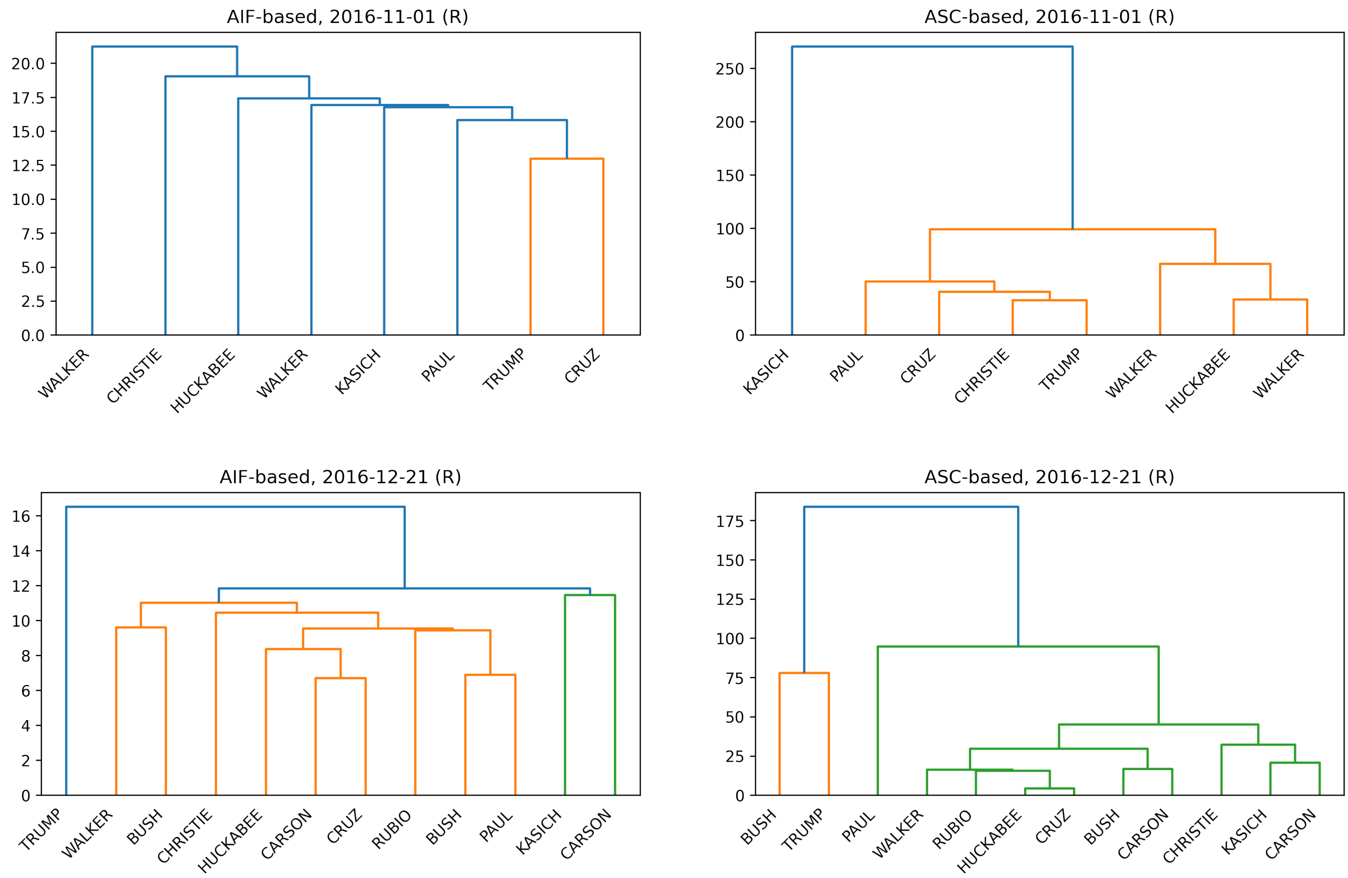


Figure 10: Extracted DRACS representations per-dimension of Republicans candidates on 2016-11-01.

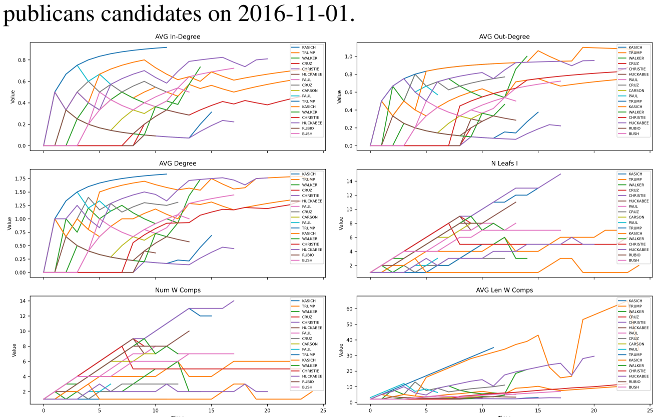
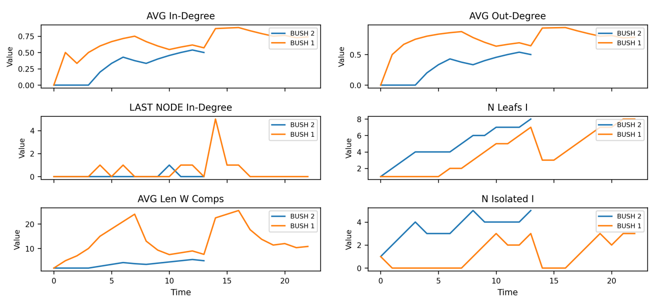


Figure 11: Extracted DRACS representations of Bushes speeches per-dimension on 2016-12-21.



tives on argumentation. It opens possibilities for detecting rhetorical tactics, understanding discourse coherence, and differentiating strategic behaviors even when surface-level argument structures appear similar.

Acknowledgements

This work has been supported in part by: the “AI for Citizen Intelligence Coaching against Disinformation (TITAN)” project, funded by the EU Horizon 2020 research and innovation programme under grant agreement 101070658, and by UK Research and innovation under the UK governments Horizon funding guarantee grant numbers 10040483 and 10055990; the ‘Artificial Intelligence for Institutionalised, Multimodal, Gamified, Mass Democratic Deliberations’ project, funded by the EU Horizon Europe Framework Programme (HORIZON) under grant agreement 101178806; the “CLARUS” project, funded by the EU Horizon Europe Framework Programme (HORIZON) under grant agreement 101121182.

References

[1] K. Al Khatib, H. Wachsmuth, M. Hagen, and B. Stein. Patterns of argumentation strategies across topics. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1351–1357, 2017.

[2] S. Arsyad, V. Widiarti, and M. F. Wulandari. Argument strategies and linguistic realizations of the discussion sections in local, national, and international journal articles in english education by indonesian authors: How do they differ and/or resemble? *ENGLISH FRANCA: Academic Journal of English Language and Education*, 5(1 May):111–130, 2021.

[3] R. Baumann and M. Heinrich. Timed abstract dialectical frameworks: A simple translation-based approach. In *Computational Models of Argument*, pages 103–110. IOS Press, 2020.

[4] R. Bergmann, M. Lenz, S. Ollinger, and M. Pfister. Similarity measures for case-based retrieval of natural language argument graphs in argumentation machines. In *FLAIRS*, pages 329–334, 2019.

[5] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *KDD Workshop*, 1994. URL https://api.semanticscholar.org/CorpusID:929893.

[6] S. Bistarelli, M. C. Meo, C. Taticchi, et al. Timed concurrent language for argumentation. In *CEUR WORKSHOP PROCEEDINGS*, volume 3002, pages 1–15. CEUR-WS. org, 2021.

[7] S. Bistarelli, M. Chiara Meo, and C. Taticchi. Timed concurrent language for argumentation with maximum parallelism. *Journal of Logic and Computation*, 33(4):712–737, 03 2023. ISSN 0955-792X. doi: 10.1093/logcom/exad009. URL https://doi.org/10.1093/logcom/exad009.

[8] S. Bistarelli, V. David, F. Santini, and C. Taticchi. Temporal probabilistic argumentation frameworks. In *CEUR Workshop Proceedings*, 2023.

[9] G. Brewka and S. Woltran. Abstract dialectical frameworks. In *Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning*, KR'10, page 102–111. AAAI Press, 2010. ISBN 1577354516.

[10] G. Brewka, S. Ellmauthaler, H. Strass, J. P. Wallner, and S. Woltran. Abstract dialectical frameworks. an overview. *Journal of Applied Logics*, 4(8):2263–2317, 2017.

[11] C. Chesnevar, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South, G. Vreeswijk, S. Willmott, et al. Towards an argument interchange format. *The knowledge engineering review*, 21(4):293–316, 2006.

[12] D. Gemechu and C. Reed. External knowledge-driven argument mining: Leveraging attention-enhanced multi-network models. In Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3688–3709, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.216. URL https://aclanthology.org/2024.emnlp-main.216/.

[13] D. Gemechu, R. Ruiz-Dolz, and C. Reed. Aries: A general benchmark for argument relation identification. In Y. Ajjour, R. Bar-Haim, R. El Baff, Z. Liu, and G. Skitalinskaya, editors, *Proceedings of the 11th Workshop on Argument Mining (ArgMining 2024)*, pages 1–14, Aug. 2024. doi: 10.18653/v1/2024.argmining-1.1. URL https://argmining-org.github.io/2024/. Publisher Copyright: © 2024 Association for Computational Linguistics.; 11th Workshop on Argument Mining, ArgMining 2024 ; Conference date: 15-08-2024 Through 15-08-2024.

[14] A. Gerdes, B. Heeren, and J. Jeuring. Properties of exercise strategies. *Electronic Proceedings in Theoretical Computer Science*, 44: 21–34, Dec. 2010. ISSN 2075-2180. doi: 10.4204/eptcs.44.2. URL http://dx.doi.org/10.4204/EPTCS.44.2.

[15] I. Habernal and I. 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, 2015.

[16] A. Hagberg, P. J. Swart, and D. A. Schult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

[17] R. Knaebel, R. Schaefer, and M. Stede. The impact of argument arrangement on essay scoring. In *Conference on Advances in Robust Argumentation Machines*, pages 147–162. Springer, 2024.

[18] J. Lawrence and C. Reed. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818, 01 2020. ISSN 0891-2017. doi: 10.1162/coli_a_00364. URL https://doi.org/10.1162/coli_a_00364.

[19] J. Lawrence, F. Bex, C. Reed, and M. Snaith. Aifdb: Infrastructure for the argument web. In *Computational models of argument*, pages 515–516. IOS Press, 2012.

[20] M. Lippi and P. Torroni. Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Technol.*, 16(2), Mar. 2016. ISSN 1533-5399. doi: 10.1145/2850417. URL https://doi.org/10.1145/2850417.

[21] L. Pragst, N. Rach, W. Minker, and S. Ultes. On the vector representation of utterances in dialogue context. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[22] C. Rapanta. *Argumentation strategies in the classroom*. Vernon Press, 2019.

[23] M. T. Sazid and R. E. Mercer. A unified representation and deep learning architecture for argumentation mining of students' persuasive essays. In *CMNA@ COMMA*, pages 86–97, 2022.

[24] R. Schaefer, R. Knaebel, and M. Stede. Towards fine-grained argumentation strategy analysis in persuasive essays. In *Proceedings of the 10th Workshop on Argument Mining*, pages 76–88, 2023.

[25] C. Stab and I. Gurevych. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 46–56, 2014.

[26] R. Tavenard, J. Faouzi, G. Vandewiele, F. Divo, G. Androz, C. Holtz, M. Payne, R. Yurchak, M. Rußwurm, K. Kolar, and E. Woods. Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, 21(118):1–6, 2020. URL http://jmlr.org/papers/v21/20-091.html.

[27] T. A. Van Dijk. Opinions and ideologies in editorials. In *4th international symposium of critical discourse analysis, language, social life and critical thought, Athens*, pages 14–16, 1995.

[28] J. Visser, B. Konat, R. Duthie, M. Koszowy, K. Budzynska, and C. Reed. Argumentation in the 2016 us presidential elections: annotated corpora of television debates and social media reaction. *Language Resources and Evaluation*, 54(1):123–154, 2020.

[29] R. C. Wilson and P. Zhu. A study of graph spectra for comparing graphs and trees. *Pattern Recognition*, 41(9):2833–2841, 2008.

[30] K. Zaczynska, P. Bourgonje, and M. Stede. How diplomats dispute: The UN security council conflict corpus. In N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8173–8183, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.716/.