Design and Evaluation of Visualization Techniques to Facilitate Argument Exploration

D. Khartabil,[1] C. Collins,[2] S. Wells,[3] B. Bach[4] and J. Kennedy[3]

[1]GFT Financial Limited, London, UK
[2]Ontario Tech University, Oshawa, Canada
Christopher.Collins@ontariotechu.ca
[3]Institute for Informatics and Digital Innovation, Edinburgh Napier University, Edinburgh, UK
S.Wells@napier.ac.uk, j.kennedy@napier.ac.uk
[4]University of Edinburgh, Edinburgh, UK
bbach@inf.ed.ac.uk

Abstract
This paper reports the design and comparison of three visualizations to represent the structure and content within arguments. Arguments are artifacts of reasoning widely used across domains such as education, policy making, and science. An argument *is made up of sequences of statements (premises) which can support or contradict each other, individually or in groups through Boolean operators. Understanding the resulting hierarchical structure of arguments while being able to read the arguments' text poses problems related to overview, detail, and navigation. Based on interviews with argument analysts we iteratively designed three techniques, each using combinations of tree visualizations (sunburst, icicle), content display (in-situ, tooltip) and interactive navigation. Structured discussions with the analysts show benefits of each these techniques; for example, sunburst being good in presenting overview but showing arguments in-situ is better than pop-ups. A controlleduser study with 21 participants and three tasks shows complementary evidence suggesting that a sunburst with pop-up for the content is the best trade-off solution. Our results can inform visualizations within existing argument visualization tools and increase the visibility of 'novel-and-effective' visualizations in the argument visualization community.*

Keywords: interaction, visualization, Text visualisation, Hybrid visualisation techniques

CCS Concepts: • Computing methodologies → Computer graphics; • Information systems → Information interfaces and presentation

1. Introduction

Arguments are artefacts of reasoning and are the core of many rational, strategic, inter-personal communicative processes including persuasion, deliberation, and negotiation. Arguments are applied in different domains to achieve various objectives. For example, in learning and education, students try to be logical and reasonable; they strive to improve their skills in arguing to convince others of their opinions and ideas [SLPM10, Sut03]. Argumentation in education and science is a collaborative discussion between two or more parties to find agreement or solve an issue [And06], [RHR17], Chapter 3]. Argumentation helps learners think deeply, provide reasoning for their thoughts, develop their collaborative ability, and share their ideas. The study of argumentation [vEGK*14] not only attempts to identify argument elements, premises and conclusions, but

also to recognize and catalogue the patterns of relationships between them [Gov13]. Studying real-world contexts of argumentative discourse, then identifying, extracting and representing the constituents of arguments and the relationships between them, in an appropriate way, is a process called argument analysis. The results of argument analysis are datasets of arguments associated with specific problem domains. Argument datasets can be used by further computational processes, for example, automated reasoning systems, and are used by people to explore and navigate knowledge domains so that the focus can lie on the specific arguments made rather than their linguistic or rhetorical presentation. The advent of automated natural-language analysis techniques in a sub-field called 'Argument Mining' promises a future in which any written text can be processed without human intervention; its arguments extracted, and stored for
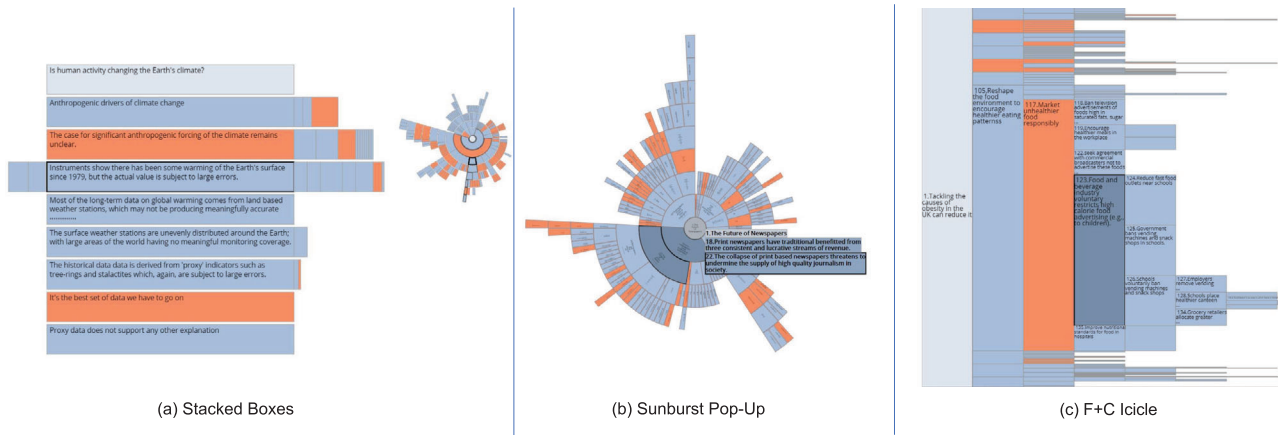
Figure 1: *Three alternative designs for argument visualization.* **Blue** *and* ***orange*** *were applied for support and oppose relations between statements in an argument, respectively.*

subsequent reuse. These argument mining techniques[SS19] are already leading to an increase in the scale of available datasets, whose limit is set by the number of computational resources that can be brought to bear rather than the availability of trained human experts. The current work is not focused on argument mining, but is driven by the need to analyse the *outputs* of argument mining, and the opportunities to do so with visualization.

Rather than understanding arguments as purely linguistic entities and relationships, they are frequently visualized using argument visualization techniques, using combinations of text, symbols, and shapes [Wal05, KBSC12] arranged in order to communicate specific information about the argument to the viewer. The main aims of argument visualization are to (1) clarify the elements of arguments, (2) identify the relationships between premises and conclusions, (3) to allow users to gain a quick overview of the domain of the arguments, and (4) to show details of interesting arguments to enable the reader to focus upon relevant parts [KBSC12].

For example, health care professionals and policymakers in 2015 tried to understand the factors behinds the increase in obesity rates in the UK [BPB15]. They were overwhelmed by the amount of data they needed to analyse and used an argument visualization tool called DebateGraph [BP08] to help unravel the complexity of the data and find solutions to tackle the obesity pandemic. Using DebateGraph assisted them in presenting the data related to the obesity issues and exploring the relations between the argument elements. Another example is an investigation into the ethical aspects of climate engineering [BC12] in which argument visualization was used to present an overview of the moral reasons for and against various actions and policies related to climate engineering.

Understanding arguments requires firstly to have a macro-structure, i.e. to see the overview of the whole argument. Second is the micro-structure that involves exploring the internal structure of argument elements. Third is the ability to read the text of the argument elements. Argument visualization tools face challenges in achieving all these requirements as the size of argument datasets increase.

Still, two major challenges in argument visualization are **(C1)** to provide an overview to present large arguments where the number of argument elements exceeds a few hundred [KLP*16], and **(C2)** due to the textual nature of the data, displaying the premises and conclusion in a readable format, in addition to the relations between them, is fundamental for users to understand the content of the argument.

A common trade-off to address both C1 and C2, is to use one of the following approaches: (a) to focus the visualization design on the structure and only show text on demand, (b) focus on the text but reduce the information displayed about the structure or (c) to provide both—text and structure—at the same time and to embed the text into the visualization, resulting in very large visualizations requiring large screens. In this paper, we propose three approaches for designing argument visualizations that seek to balance the detail of textual content against the overall structure (Figure 1). Our first technique (**Stacked Boxes**) focuses on the text, indicating connected and related argument elements through visual cues, with a linked miniature sunburst providing a structural overview. The second technique (**Sunburst Pop-Up**) focuses on the structure using a sunburst layout adapted to suit argument criteria and overlays text on demand using a modified labelling technique. Our third technique (**F+C Icicle**) is an adaption of an icicle plot to visualize the structure while using focus and context to enlarge the premises in a focused node chain [CK08]. We conduct a controlled user study to collect qualitative and quantitative feedback on the three techniques. Also, we interviewed argumentation experts to collect feedback from the perspective of those who are actively working in the argumentation domain. To reflect the outcome of the evaluation, we designed a hybrid technique that combines the Stacked Boxes and Sunburst Pop-Up.

2. Background and Related Works

2.1. Argument structure and visualisation

At their core, a single argument is a simple textual structure that comprises one or more premises which support a conclusion. The
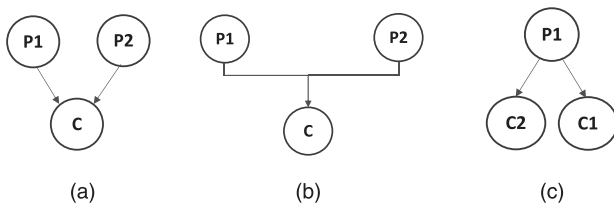
Figure 2: *Standard argument structures [Gov13]: (a) convergent arguments; (b) linked argument; (c) divergent arguments.*

way in which premises support a conclusion gives rise to three basic structures; (i) *convergent* (ii) *linked* and (iii) *divergent* arguments, each of which has a standard diagrammatic representation (Figure 2). In a convergent argument, the premises (p1, p2) are connected to the conclusion (c) with each premise working independently to support the conclusion as shown in Figure 2(a) (similar to a logical OR). In a linked argument, as illustrated in Figure 2(b), the premises (p1, p2) work together to jointly support the conclusion (c) (similar to the logical AND). In a divergent argument, a premise (p) supports multiple different conclusions (c1, c2), as illustrated in Figure 2(c). Note that in the case of both convergent and divergent arguments, the argument can be split into individual sub-arguments [Gov13]. Individual arguments are frequently chained together, with the conclusion of one argument acting as the premise of another argument, yielding potentially large and complex chain, tree and graph structures.

A challenge in visualizing arguments at scale is the determination of which elements to include and which to exclude. This can be seen as a trade-off between aspects of the visualization that give clarity at different levels of granularity. For example, when attempting to visualize an overview of the domain of an argument dataset, the fine-grained structure of each argument might be omitted so that more arguments can be viewed at once. As a result, certain judicious decisions have been made in this paper with respect to the granularity of argument structure, namely; (1) the final conclusion is the root node, (2) linked and convergent arguments treat premises as children of a conclusion and (3) for divergent arguments, we duplicate the premises in order to create a strict tree structure.

2.2. Argument visualisation

Argument mapping and argument diagramming are used interchangeably to refer to the process of moving from a purely linguistic argument to a visualization that makes aspects of the structure and content clear. There have been many approaches to visualizing arguments, including the standard diagram of Araucaria [RR04], which implements a form of argument map found throughout the philosophic and communication theory literature as well as also supporting the approaches of both Toulmin [Tou58], whose layout of argument is popular in critical thinking and philosophy of science, and Wigmore [Goo00], whose charts are used in historic legal argumentation. Furthermore, many online discussion platforms increasingly attempt to visualize discussions, at varying levels of complexity, to aid their users in navigating complex domains of conflict[SGP13a, AWK*18].

Argument visualization tools can be categorized based on their layout by describing their visual techniques and highlighting their limitations. Examples of argument visualization tools are illustrated in Figure 3. First, *indentation layouts* emphasize hierarchical structures by placing premises vertically at varying levels of indentation [LS13, Ver03, IQDLBS16, Kle12]. To provide users with more details about the arguments, some tools used two or more views such as Hermes [KP01]. Indentation layouts are familiar to users due to their similarity to file browsers. In these layouts, text can occupy nodes without crossing (C2). However, they fail to provide an overview of the structure in large arguments due to the limitation of the window size. Also, indentation layouts can be overwhelming due to the number of expanders needed to get to specific data [FNS13].

Second, *node-link layouts* present a tree structure or network based on the argument structure [PEAS*18, RMRW06, VDBVP07, IQDLBS16, UDLB19, WD17, CH19, KRW*21]. DebateGraph uses a node-link layout with various views [BP08]. The first one is a simple triangular vertical tree with the root on the top and child nodes on the bottom. The second is a radial tree where the root is located in the centre, and children are placed on circles surrounding the root according to their depth in the tree. Rationale [van07] proposed a new tree structure called hi-tree. The key novelty in this representation is to allow statements that are linked together and at the same level to be grouped into "compound nodes" instead of using links to save space. Node-link layouts are useful in showing the supporting or opposing relations between argument elements in a simple way.

However node-link visualizations do not tend to scale well as datasets grow larger, particularly if the same level of detail is maintained across the visualization. They also tend to require a larger visualization space, and in the worst case a larger screen, to effectively see an overview of the domain of the argument. We can see this in many of the existing argument analysis tools, including not just DebateGraph [BP08], but also Araucaria [RR04], OVA/OVA+ [SGP13b], Rationale, and MonkeyPuzzle [WD17]; as the amount of argumentative information to display in the visualization grows, the visualization becomes more cramped, and especially in those visualizations that are graph-based, more complex. The use of zoom is not an effective solution as when zooming out we lose detail, hence we explored alternative visualization techniques for the problem of surveying and overviewing larger datasets. Node-link layouts have a long history in argument visualization, at least back as far Wigmore's [Goo00] 1873 legal argument diagrams, and yet in the intervening period the cutting edge of dealing with more data in the visualization has progressed from Wigmore's encouragement to "sharpen your pencil" in order to draw smaller diagrams, through to the zoom facilities available in the aforementioned argument.

Third, *nested layouts* overcome the problem of wasted space in the node-link layout. For example, SenseMaker [Bel97] uses a box-based Venn diagram to visualize small hierarchy data [JS91]. It displays conclusions in frames with evidence denoted by dots located in the related frame. It is similar to the treemap [Shn92], but blank spaces between the child nodes are allocated. Issue Maps uses a treemap layout to solve the problem of readability in large-scale arguments [BM12]. Rectangular blocks with different colours and sizes are used to present arguments. Instead of displaying
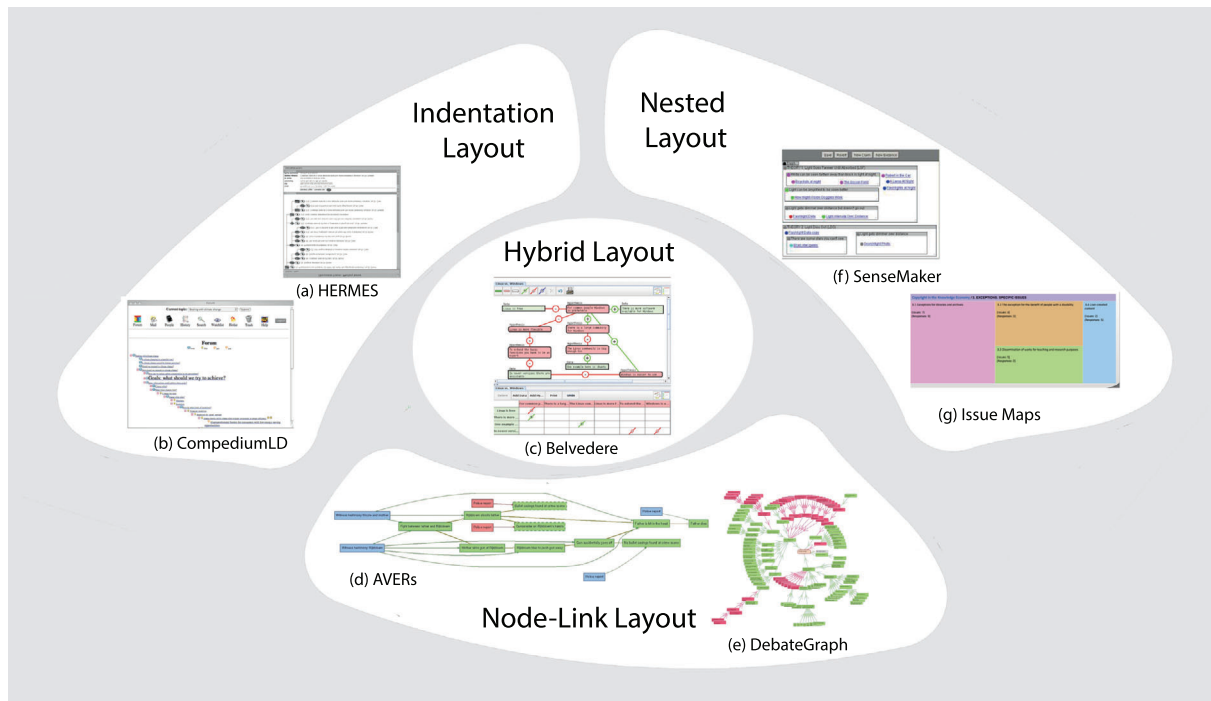
Figure 3: *Snapshots of some existing argument visualization tools: (a) HERMES [KP01], (b) Collaboratorium [KI08], (c) Belvedere [SWCP95], (d) AVER [VDBVP07], (e) DebateGraph [BP08], (f) SenseMaker [Bel97], (g) Issue Maps [BM12].*

the children inside the parent as rectangles, Issue Maps shows only the number of children, and thus, the full structure is not presented.

Finally, hybrid layouts combine two layouts to achieve the balance between providing an overview and reading the text, e.g. Horn uses treemap and node-link techniques to present the arguments [HC98]. Belvedere [SWCP95] applied a matrix layout as an additional presentation alongside a node-link graph to help users explore the argument relations. Kialo, a tool for critical thinking and serious discussion, uses a sunburst layout to show an overview of the data, and when the user clicks on the discussion, it moves to the node-link layout to reflect the local structure, with a linked table to present the text of the arguments, however no study of the tool has been undertaken [Pit07]. Ullmann et al. propose the *Collective Intelligence* dashboard, a variety of graphical representations of arguments, for example treemap, conversation network, and zoomable nested circles [UDLB19]. The views, composable in a coordinated view, showed promise in a small-scale study to improve the understanding of argument in a large-scale discussion. Rather than creating an analytic dashboard, our work is focused on the specific issue of balancing structure visibility and text readability in a single visualization.

The recent argument mining tools use visualization to help explore and understand the argument mining datasets. Gold et al. proposed novel interactive visualizations to analyse the deliberation [GEAHJ*17]. Different types of visualizations were used to demonstrate the topic distributions and to give an overview of the argumentation structure. VIANA is an interactive annotation sys-

tem for augmentations. It identifies five high-level analysis tasks, which are close reading and note-taking, annotation of arguments, argument reconstruction, extraction of argument relations, and exploration of argument graphs using node-link layout [SSKEA19].

The zoom and pan technique is used in argument visualization tools to increase the size of the focused point and navigate from one focused point to another. Tools such as OVA [SGP13b] and Argunet [SVB07] have used an overview and details technique, where detailed information is displayed beside an overview of the entire view in a small rectangle at the corner of the same window. The small view of the overview is separated from the details view in the Reason!Able [vG03]. While using this technique helps the user to understand more information about the data, sharing the display area can present a memory burden for users [Mun14].

DebateGraph uses a technique for supporting interactive exploration developed by Yee [YFDH01]. The layout of the tree animates into a new layout, and the focused node is located in the centre. However, this technique changes the geometry between the views, which may confuse users and make it difficult for them to understand the relations between the views.

To conclude, many argument visualization tools have been proposed to help people in understanding arguments, exploring their underlying structure and visually mapping the flow of inferences. However, these tools face the challenge of readability and scalability when they are presenting over a hundred nodes, and have difficulty in presenting nodes containing lengthy text. Therefore, new techniques and tools are required to address these challenges.

2.3. Space-filling visualisation techniques

The main challenge that argument visualization layouts face is the space when arguments grow in number (C1). Space-filling techniques use juxtaposition to express the relations between the nodes instead of links to save space and condense the overview. These techniques can be rectangular like a treemap [Shn92] or icicle [KL83], or circular like a sunburst visualization [SZ00, CCP09]. Sunbursts have an explicit structure compared with treemaps and are ranked highest in aesthetics and one of the top-performing visualizations regarding efficiency and effectiveness [CM07]. Icicle plots and sunbursts are more suitable layouts for providing an overview than treemaps [BNK16, WYM19]. The structure in the treemap is not as clear as in the classical tree [War12]. Therefore, icicle and sunburst layouts were considered promising solutions for providing an overview of argument structure and its navigation.

The method used to explore the text and read argument contents in the literature, for example labelling, is discussed in the following section.

2.4. Labelling in visualisation

Labelling in information visualization reflects the relations between visual objects and textual elements (C2). Based on the location of a label relative to the visual object, we can classify labels into internal and external labels. Internal labels are located inside the visual object. This way of labelling is explicit, readable and points immediately to the visual object. The problem with this type of labelling results from overlapping of labels due to the lack of space when the text is long [HGAS05]. An external label is located outside the visual object it refers to and is connected using a leader line. Many algorithms are designed to optimize the location and alignment of external labels [FHS*12]. Boundary labelling is a type of external labelling designed for long labels, where a rectangle is used to frame the label(s), a leader connects each label to the object, and one or more labels can be placed in one rectangle [BKSW07]. Excentric labelling is a dynamic technique of boundary labelling, revealing labels of items near a point of interest [FP99] which has been further improved through extensions [BRL09, FHS*12].

When designing labels, features such as readability and aesthetics should be considered [HGAS05], and we should avoid labels overlapping with each other or with the related structure. These problems are increased when the number of labels rises, or when labels become long [OJP14].

3. Methodology

Several models for designing visualization tools have been proposed in the literature [CMS99, HA06, SP06]. One of the most popular models is the nested model [Mun09], which divides the visualization design into levels of abstraction. We applied the main steps of the nested model in the context of the argumentation domain as follows. First (1), we reviewed existing work on argument domains to build an understanding of argumentation. Then (2), we conducted interviews with argument experts to understand the issues with existing approaches and the requirements for improved argumentation tools. The experts were chosen from various argumen-

tation domains, who have used or been involved in designing one of the existing argument visualization tools. Then (3), we mapped the data collected from the interviews into information visualization vocabularies; arguments have a graph structure consisting of nodes (argument elements) and relations between them. Finally (4), we created encoding and interaction, according to the third step in the nested model. Argument elements (premises and conclusion) were encoded as nodes in a variety of tree representations. The colour encoding of the nodes reflects the relationships between them, with red nodes *opposing* their conclusion and green nodes *supporting* their conclusion. Interaction methods customized to each encoding were provided to navigate the text and read the arguments. To validate the second and the third steps in this model, experts were contacted to test the techniques and give feedback. –

4. Requirements and current limitations

To better understand how to support experts and collect the user requirements, we conducted semi-structured interviews [IZCC08] with seven experts. The experts were researchers, senior researchers and associate professors in different areas of argumentation, for example argument mining, problem-solving, and argument visualization. Details about the experts are presented in Table A1.

Each interview lasted between 30 and 45 min. We divided the interview into three sections as in Table A2. The first section covered the use of previous argument tools and the challenges the experts faced using them. The second section had questions regarding the size of the data they used. The third section was composed of questions regarding user requirements and the way experts read and attempt to understand arguments. During the interviews, experts mentioned the following limitations:

L1 'Most of the tools used nodes and arrows, which is brilliant when you have a small number of arguments. But, when the number of nodes increases to 20–30, the diagram becomes very dense'. For example, 'I used a lot of the argument tools like Araucaria, DebateGraph, OVA, and Rational, the graphs becomes dense and impenetrable with large-scale data'.

L2 'The biggest challenge of presenting the large-scale of arguments is that we want to see the whole picture (how the argument elements connect with each other, which argument has the most support/oppose statements) and see the details using the same tool.'

L3 'The tools we have designed are only for small scale arguments; I would like to have tools that handle a large number of arguments.'

All the experts raised concerns over the capability of node-link layout in visualizing dataset of more than 50 nodes especially with the text presented inside the nodes.

By analysing the interviews, we conclude that the ability to provide a quick summary of the overall data and read the text were key priorities. We define user requirements R1–R4 as follows:

R1—Show argument overview: Present the entire argument elements, all at once, to know how they are connected to the main topic and with each other. Also, explore the logic flow

of the argument elements in the proper order making the reasoning structure explicit.

R2—Read the text inside the node: Read the text from the conclusion to all the statements that support/oppose this conclusion to help understand them.

R3—Provide navigation: Support experts to keep track of where they are relative to the overall argument structure.

R4—Enable comparison: Users need to be able to compare two chains of argument elements.

5. Designing Argument Visualisation Techniques

To satisfy our requirements R1–R4, we designed several techniques and implemented prototypes through iterative refinement. This section describes the design of our three techniques and their prototypes.

5.1. Initial prototype

The design of our initial prototype is mainly based on findings from the literature, our requirements R1–R4, and challenges C1 and C2. Showing the data structure is an indispensable requirement for understanding arguments (R1). Space-filling techniques are promising solutions to present the argument structure as they are effective at displaying an overview of the whole data. Therefore, our first prototype was based on the sunburst layout, adapted to

1) deal with convergent and linked arguments,
2) visualize opposing and supporting arguments, as well as
3) to provide for good readability of the text of individual premises and conclusions.

In the sunburst, the main conclusion of the argument is placed in the centre of the visualization (the root node), with the premises branching outwards. The colour encoding of the nodes (premises) reflects the relationships between them, with **red** nodes *opposing* their conclusion and **green** nodes *supporting* their conclusion. This scheme is based on the established colour encoding in argument visualization tools. We recognize that, while red and green may carry helpful semantic connotations, this colour scheme is not accessible for users with common forms of colour vision deficiency. This is addressed in later prototypes which use accessible palettes [HB03] as shown in Figure 1. The text for argument statements is presented inside each sunburst node (premise), leaving the text wrapped, and the font size changed based on the length of the containing text (R2, C2). Despite showing the argument structure efficiently, the size of the sunburst nodes does not allow the full presentation of long text, especially for premises being located far away from the tree centre because of their reduced size. As a solution, the full-length statement of the node is popped-up while hovering over the nodes in the sunburst as displayed in Figure 4, allowing for quick navigation through the nodes and reading the text (R3).

This prototype has clear limitations such as reduced readability of text and nodes getting very small for premises deep inside the tree. Consequently, we proposed a coordinated view consisting of a sunburst layout to present the overview with a node-link layout to help navigate through the text as illustrated in Figure 5. When the users click on any node in the sunburst layout, the node-link
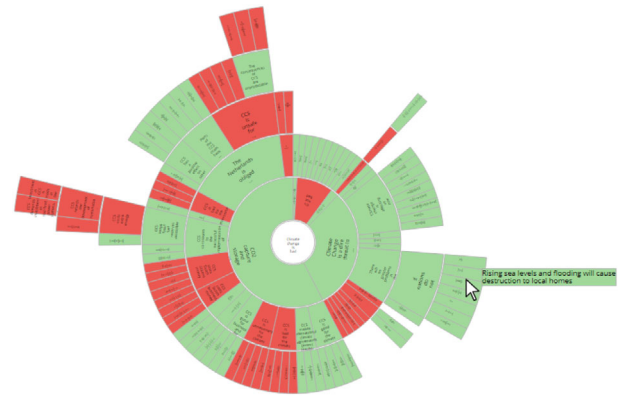

Figure 4: *The sunburst with colour encoding, red nodes opposing and green nodes supporting.*
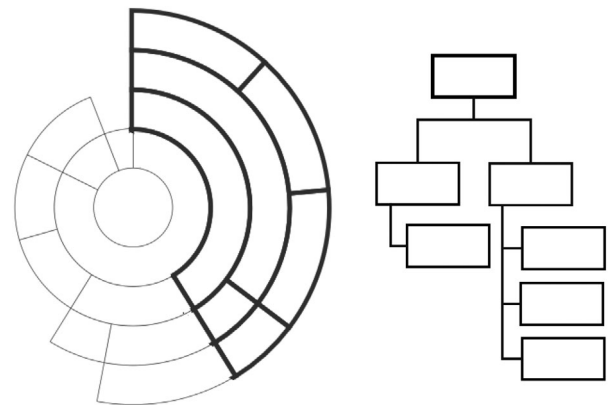

Figure 5: *A coordinated view consists of a sunburst layout to present the overview on the left with a node-link layout to help navigate through the text on the right.*

layout is displayed in the coordinated view to reflect the sub-tree from the focused node to all its children. The argument sub-tree in the sunburst is highlighted to reflect the node-link view. For quick navigation through the nodes and reading the text, the entire label of the node, i.e. full-length statement, is popped-up through hovering over the nodes in the sunburst.

However, this prototype inherits the limitation of the node-link layout in terms of space-efficiency. In addition, it is challenging to show the focused node's children while keeping the labels readable, especially when the number of nodes exceeds about 40 nodes. The pop-up label in the sunburst allows the users to read the text, but it is limited to showing the text of the hovered node only. Therefore, going through an argument's chain requires a lot of navigation.

To overcome these limitations, we designed three techniques which are inspired by the previous prototypes and which explore the very large design space around combining hierarchy visualizations (see Section 2.1) while allowing for effective reading of arguments. Figures 6, 10, and 13 illustrate the development of all three techniques. The grey scheme reflects the hierarchy structure, that is the root is coloured with a dark grey that shades lighter by level towards
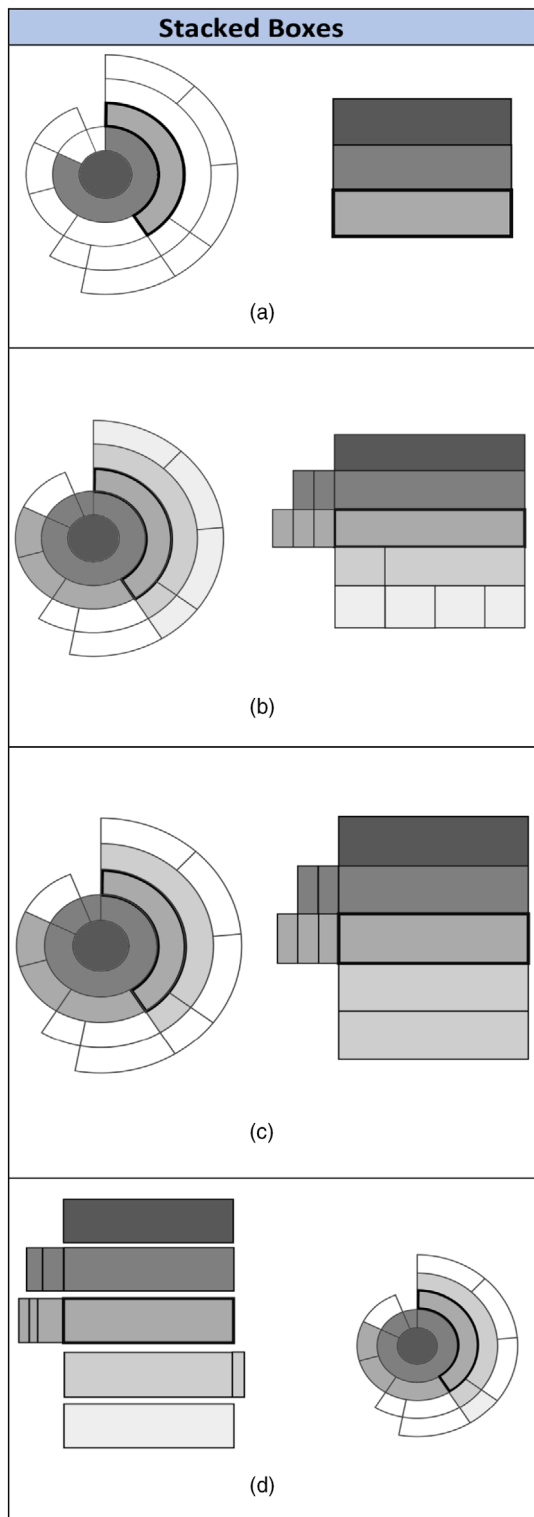
Figure 6: *The iterative design sketches of the Stacked Boxes techniques.*

the leaf nodes. We coloured nodes in the sunburst to show the nodes in the coordinated stacked box or pop-up label and demonstrate the difference between the versions.

5.2. Stacked Boxes–Stacked boxes with sunburst

Stacked Boxes (Figure 6(a)–(d)) are two juxtaposed coordinated views with a sunburst visualising the argument structure and a set of stacked boxes showing the chain from a premise to its final conclusion (via intermediate conclusions/premises); in other words the path from a child node to the root node in a tree.

Stacked Boxes passed four cycles of design development. Our first version represents the list of premises from a selected premise to the root in a simple fixed view shown in Figure 6a. This version helps the user to read the argument text quickly from the focused node to the root conclusion, but it does not enable reading the content of the focused node's children or sibling nodes. Also, it is difficult to read the text in one view and navigate in the other.

In the second version, this box layout was improved to offer quick navigation and reading of the sibling nodes on the same level as displayed in Figure 6b. This version displays the focused node, its parents, siblings and all the children in its sub-tree as smaller boxes to the side (siblings) and the bottom (children). This version provides intuitive navigation through the tree structure by clicking on any of the boxes. However, where there is more than one child node of the focused node, the user cannot read the text inside them because the box's size is limited.

In the third version, we aimed to reveal the text of the focused node and only its immediate children as *vertically aligned boxes* (Figure 6c). However, we found that this approach resulted in a mixed visual metaphor which could be confusing for the user as the layout stacks nodes vertically from the root down to the focused node (as per the tree), while below the focused node the vertical stack represents the first level children of the focused node (a horizontal layer in the tree).

In our final version of Stacked Boxes (Figure 6d), we treated each layer of boxes as a navigable strip of siblings which present the focused node highlighted, parents (stacked above), first child of each level in the tree below the focused node, with the siblings navigable for all levels. When the user flips through the sibling nodes (clicking a box or using arrow keys), the selected node becomes the focused node with no changes on the upper level, while underneath, the focused node changes to reflect the chain of this node. A small space between the levels is created in the Stacked Boxes to show that each level is not directly connected with the level above or below except in the focus stack. The subtree of the focused node is highlighted in the sunburst visualization (R3).

As the Stacked Boxes became the focus view for reading the text, we changed the layout to make the Stacked Boxes the main view on the left with the sunburst reduced to a small overview figure on the right for viewing and navigating the overall structure (Figure 7).

We offered a search bar for the user to search for specific terms in the statements. Search results are emphasized using a darker shade of the original nodes' colour, as illustrated in Figure 8. The idea
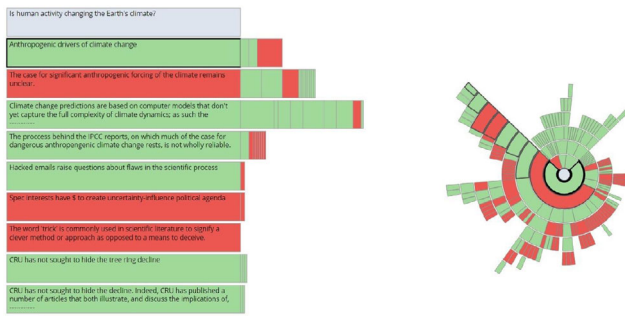
Figure 7: *On the left for reading the text with coordinated sunburst overview on the right for navigating the structure.*
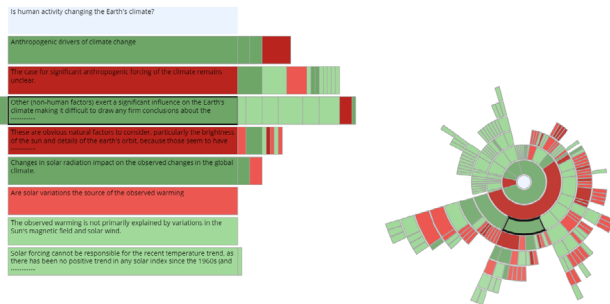

Figure 8: *Search results in the Stacked Boxes and the sunburst. The results are highlighted using a darker shade of the original colour of the nodes.*

Figure 9: *Linking arguments in Stacked Boxes by introducing a dashed vertical line. The linked argument of interest is placed at the bottom, so its children (if any) are displayed underneath it.*

structure twice: as sunburst overview and as previews integrated with the stacked boxes.

5.3. Sunburst Pop-Up—Sunburst with stacked pop-up labels

Inspired by our solution of the stacked boxes, we wondered if we could integrate these boxes more tightly with the sunburst visualization (Figure 10a–d). Sunburst Pop-Up is a sunburst, overlaid with stacked boxes as used in the first iteration of the stacked boxes, limited to only showing the parents of the focused node (no children or siblings). However, due to the potential size of the stacked pop-up labels, the design of the placement of the label near the hovered node occludes part of the tree structure. We proposed the following four options for positioning the labels (Figure 10a–d):

1. **Label-1** has a *fixed position* for the stacked pop-up labels, close to the centre of the sunburst on the opposite side of the window to where the mouse is positioned. It lists the labels in the same order as the argument elements are visualized in the sunburst (Figure 10a). The stacked pop-up labels always present the nodes from the centre to the hovered node, which means the reading order of the list can flip between the top and bottom halves of the circle.
2. **Label-2** shows the stacked pop-up labels close to the hovered node and in the same order as the argument elements are visualized in the sunburst, that is, top-down or bottom-up (as in **Label-1**) (Figure 10b).
3. **Label-3** always displays the argument elements from the root node down to the hovered node while the stacked pop-up labels is placed outside the sunburst and close to the hovered node (Figure 10c).
4. **Label-4** shows the argument elements from top-down to the hovered node. Based on the mouse position in the window and the size of the stacked pop-up labels, the list is displayed in a location inside the window (Figure 10d). The position of the stacked pop-up labels is dynamic depending on whether the hovered node is close/far to the centre.

Upon implementing these four options, we found that the list could easily go outside the window border in **Label-2** (Figure 10b) and **Label-3** (Figure 10c). Also, in **Label-1** (Figure 10a), the

behind keeping the colours is to allow the users to identify the nodes as supporting or opposing. Because the layout of the Stacked Boxes does not display the full argument structure, some search results may not appear in the Stacked Boxes. Thus, we provide a shortcut (Alt + S) to step through the search results; related nodes are pulled into the Stacked Boxes as the results are cycled through.

We also set the width of siblings to reflect the number of children in the whole sub-tree instead of using a fixed width (Figure 6d). However, this could occupy a lot of space due to the potential number of children; therefore we changed the width to reflect the number of children only on the next level of children, calculated as follows: Nodes with no children have a fixed size equal to *Size*. Nodes with n children have the $Size_n = (n + 1) \times Size$.

To distinguish the linked arguments from convergent arguments (Figure 2), we introduce a small bridge that connects linked arguments as part of the nodes in the sunburst (Figure 9(r)). Linked arguments are presented in the Stacked Boxes as sibling nodes. As it is important to the user to read the texts of the linked arguments together, they are displayed in a single node separated by a dashed line, as illustrated in Figure 9(l). The linked argument of interest is placed at the bottom, so its children (if any) are displayed underneath it.

In summary, we believe Stacked Boxes offers quick and fast navigation through the tree structure while clearly showing the tree
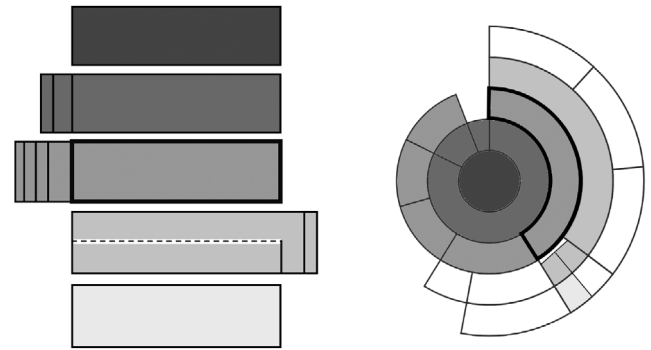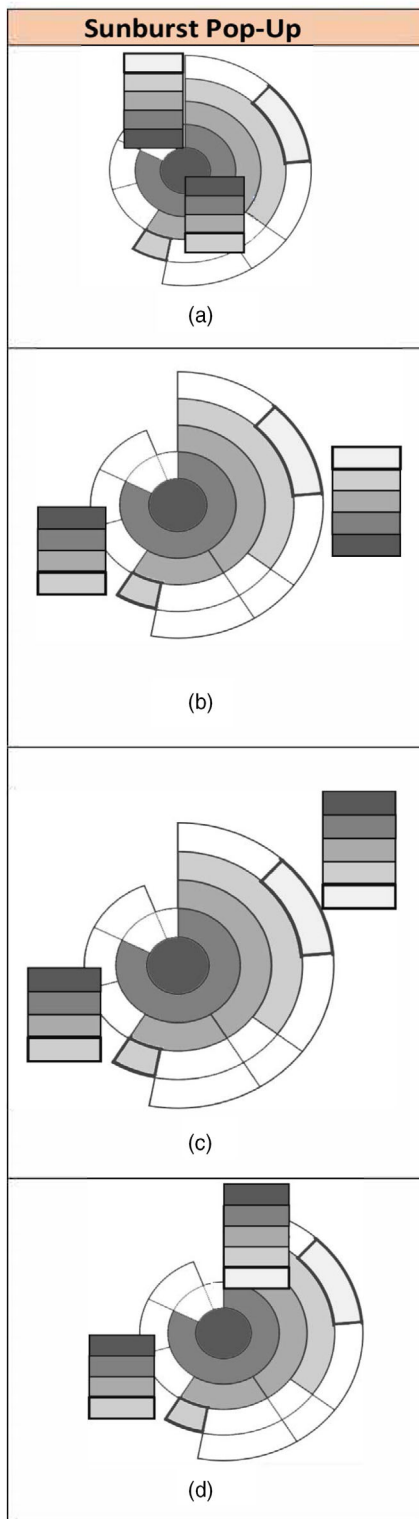
Figure 10: *The iterative design sketches of the Sunburst Pop-Up techniques: each with two examples of hovered nodes and stacked pop-up labels to show the label placement strategies.*
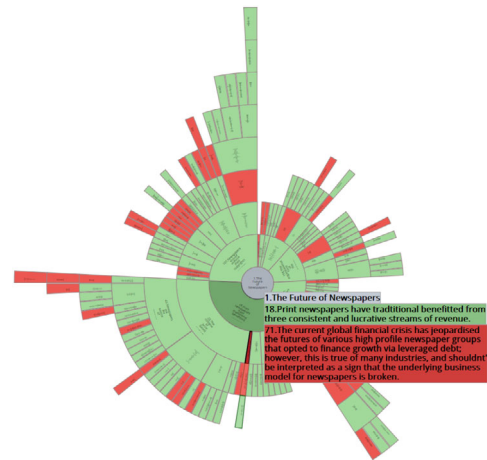

Figure 11: *Sunburst Pop-Up with label boxes showing all the highlighted argument's parents until the conclusion (gray box).*
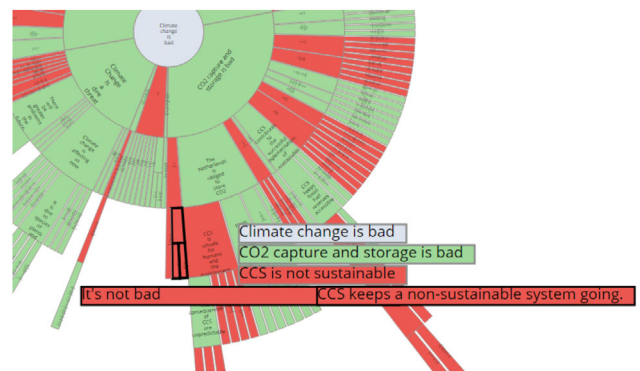

Figure 12: *Linked arguments in Sunburst Pop-Up.*

location of the stacked pop-up labels is presented far from the hovered node, especially when the node is far from the centre of the sunburst. In **Label-1** and **Label-2** changing the reading order from top down to bottom up could be confusing. For this reason, we chose the **Label-4** (Figure 11) where the stacked pop-up labels is located inside the window. The linked arguments (to distinguish them from convergent arguments) are again displayed with a bridge node which joins linked nodes at the same level in the sunburst (Figure 12).

In summary, we found that stacked pop-up labels provides quick navigation and fast hovering to read the text.

5.4. F+C Icicle—Focus and context with icicle plot

The stacked boxes in both previous techniques made us think of icicle plots which provide good space for text (for the nodes close to the root node) while providing for tree structure visualization. Thus, as a third design, we adapted the icicle plot with a focus and context (F+C) (Figure 13a–c).

In our design, the root node is located at the left-hand side, while child nodes stretch towards the right. We choose this orientation of
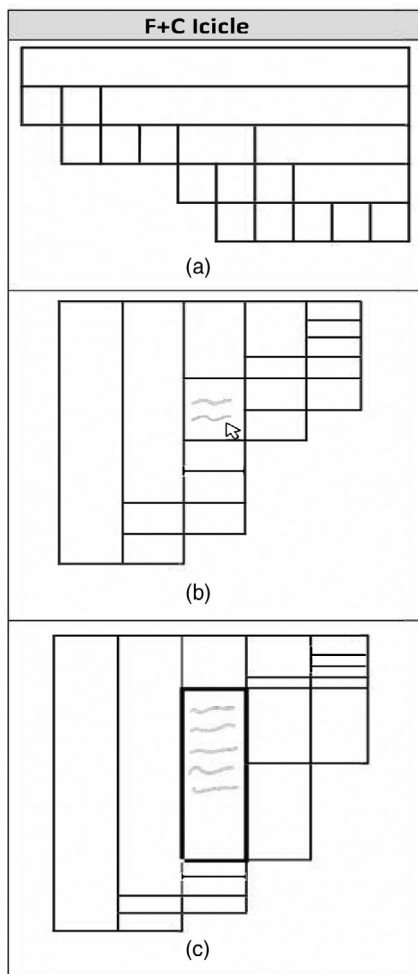
Figure 13: *The iterative design sketches of the F+C Icicle technique.*

Figure 14: *(a) The overview of the Icicle layout; (b) The F+C Icicle technique expanding one chain of the argument.*

Figure 15: *Linked arguments in F+C Icicle.*

the icicle to present the argument chain horizontally, which is more natural for reading and also gives more space for the text than the orientation in (Figure 13a). The text is truncated, wrapped and has a font size that changes based on the size of the containing node. To allow users to read the chain of the argument, we applied a F+C technique when a user clicks a node in the icicle plot (Figure 13b). It enlarges all parent arguments (to the left) and children. It also scales all expanded argument boxes up to a size that fully shows the contained argument text as shown in Figure 13c. At the same time, the text of the sibling nodes of the clicked node is displayed and enlarged, allowing users to navigate through the siblings quickly. The text of all the other nodes are removed as there is no space to present them, and they would be unreadable as displayed in Figure 14. If two or more arguments are linked, an additional box is introduced that bridges the boxes of these linked arguments (Figure 15).

5.5. Feedback

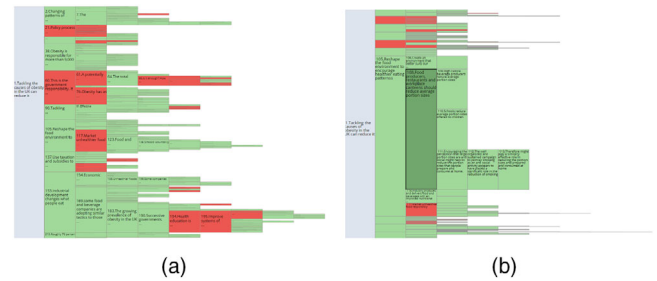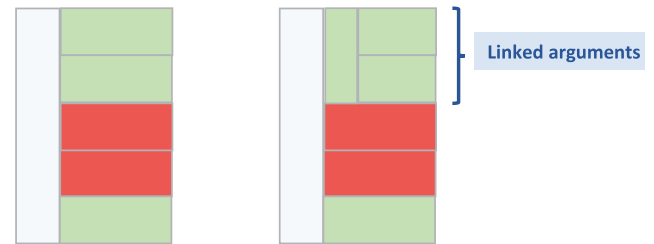The argument experts who were involved in the interviews of the users' requirements were contacted, with respect to the Stacked Boxes and the Sunburst Pop-Up, and one expert accepted to volunteer to get informal feedback. She tried both of the techniques, and provided her feedback that is listed below:

- I think overall this is a very helpful visualization for quickly getting a sense of the argument tree and the most significant topics."
- "I can easily distinguish the pros and cons."
- "It's great that you outline the part of the diagram that the comments apply to in the left panel."
- "I like the text list (stacked-boxes), and the navigation using the keyboard is fast and easy".

In addition, we did another round of informal evaluation with the expert (who was involved in the user requirements stage), when we finished the three prototypes and the following feedback was provided. "I liked the StackedBoxes in the way that you present the whole chain of the argument, and at the same time, it allows us to go through the content of the children in an easy way. The technique you used in Icicle F+C is good that it presented all the content of the clicked node, but I lost the overview in the layout. In my opinion, combining between two views of visualization, one for structure and one for the content is a promising solution to solve the problem of the large arguments. For the content, I like the linear layout which is almost similar to your StackedBoxes".

6. Controlled Usability Study

Our study aimed to compare effectiveness and efficiency (time and errors) for our three techniques, that is Stacked Boxes (Figure 6d), Sunburst Pop-Up (Figure 10h), F+C Icicle (Figure 13k), and answer the following questions:

Q1: Which layout (sunburst or icicle) best supports overview of the argument structure?

Q2: Which technique (Stacked Boxes, Sunburst Pop-Up, and F+C Icicle) best supports reading *argument text*?

Q3: Do coordinated views (Stacked Boxes) improve or degrade the performance (time and error) compared with to the integrated views (Sunburst Pop-Up and F+C Icicle) on arguments visualization?

Based on these questions, our hypotheses were as follows:

H1: Sunburst layout (Stacked Boxes, Sunburst Pop-Up) performs better (time and accuracy) than an icicle plot (F+C Icicle) in showing argument structure.

H2: F+C Icicle is better (time and error) than the Sunburst Pop-Up in reading the text; F+C Icicle provides a better technique for reading the text than Sunburst Pop-Up as the position of the stacked pop-up labels is not fixed.

H3: The hybrid technique (Stacked Boxes) is more accurate (time and error) than the other two techniques to achieve the experts' requirements; it allows the user to read the text carefully and navigate through the siblings, while, the sunburst shows the overall structure of the argument.

6.1. Participants and setting

We invited 21 participants for the first stage of evaluation (13 males, eight females), aged 25–37. They all had normal or corrected-to-normal vision without any colour impairment. Twelve of the participants were PhD students in computing or engineering, and nine were working as researchers. None of the participants had any background in argument visualization. We did not focus on expert users because they are busy and not easily available, making it difficult to get a large number involved in the evaluation. In addition, different domain experts have non-overlapping foci that are pertinent to their own domains but which are not always common to all domains of argumentation. For example argumentation theorists might focus on instances of particular schemes, whereas legal argumentation scholars might focus on links from specific arguments to case-law, and educational users might focus on specific pedagogic techniques. Instead, our focus is upon those features of argument visualization that are pertinent to exploration of argument domains regardless of topical focus. The study was run on the Intel Core i7 and 15-inch flat screen with a resolution of 1920 x 1080 pixels. During the study, we collected information about the techniques' performance, usability, and interactive features based on one-to-one sessions. The data analysis was performed using Holm's Sequential Bonferroni correction [Abd10].

6.2. Tasks

We generated three representative tasks that aimed to test different aspects of argument understanding based on the requirements described in Section 4, which are overview and details, showing the argument elements, and navigation. One of the experts involved in the user requirements was contacted and our tasks were discussed with him. Each task has a corresponding question in the study.

- **Content and Structure (CS):** In this task, which is related to requirements R1 (seeing the overview), R2 (reading the text), and R3 (navigating), we asked the participants to *'find the reasons or explanations that support or oppose a given statement'* related to the topic. The participant needed to look for a specific argument element and read the reasons connected with this argument to find the answer (content). For this task, we had two questions. One related to a statement near the root and the second question was associated with a statement near the leaves of the tree. These two questions explore the difficulty of navigating nodes in the sunburst and the icicle layouts when nodes are far away from the centre.

- **Content (C):** Here, we asked the participants to *'look for all those statements which include <a specific word>'*, for example 'energy'. The participant had to enter the search term in the search box which would result in relevant nodes being highlighted. To force the participants to read the text, we presented the search results as follows. When you look for a specific word like 'eat', all the words that contain this keyword like create or treat will appear in the search results and we informed the participants about that. Only by reading the text, the participants will distinguish between relevant and irrelevant results. This task is connected to requirements R2 (reading the text) and R3 (navigating).

- **Structure (S):** In this task, which addresses requirement R1 (logic flow of the argument), the participants were asked about the structure of the arguments. The participants should survey the whole structure and check the relations between the argument elements to find the correct answer. To examine that the structure is clear, the participants were asked not to count the nodes on the graph.

6.3. Dataset

We choose three common topics that did not require any expert knowledge to understand: climate change, the future of printed newspapers, and obesity. The datasets were collected from Debate-Graph [BP08]. Our data included 202 nodes and 201 relations for climate change, 200 nodes and 199 relations for the future of printed newspapers, and 221 nodes and 220 relations for obesity. The relations have two types, opposing and supporting ones between the statements in the hierarchy structure. The statements can be convergent or linked as described in Section 2.1.

6.4. Procedure

Pilot studies with three volunteers were undertaken to check the timing and difficulty of the study, before the evaluation with real participants was started. No pilot participants had any background in argument visualization. Upon completing this pilot, the questions and instructions of the study were updated to improve their clarity (final versions appear in this paper).

Each session of the evaluation lasted between 30 and 60 min and started with an explanation of the aim of the study and clarification of how each technique works. Three trial questions, one for each technique, were given before starting the study to allow the participants to become familiar with the techniques and raise questions. We provided participants with printed instructions regarding the three techniques, which they could keep during the study. The

study consisted of three blocks. Each block presented a technique with four task questions representative to our tasks CS, C, and S, followed by three questions about the subjective usability of the prototype. All questions are listed in appendix Tables A4 and A5. Before seeing the visualizations and carrying out the task, participants were first prompted with the question and offered time to read it. After they understood the questions, participants confirmed that they were ready to start the evaluation by clicking on the start button to access the visualisation. Clicking on the button would activate a timer to measure how long it took participants to find the answers. To avoid the influence of differing typing speeds, the participants could right-click on any node and the answer, i.e. the contents of that node, were transferred immediately to the answer box.

To minimize the impact of the technique and dataset order and combination, all the possible combinations between the three datasets and three techniques were generated and then assigned in random order to the participants. All sessions were recorded using a screen and audio recorder. At the end of the session, participants were asked to rank the techniques based on their preference, to provide the reason(s) for their selection, and to provide general comments or feedback.

7. Results and Discussion

To decide which statistical tests were required, we applied the selection process of Marusteri and Bacarea [MB10], using Shapiro-Wilk to test for normal distribution. We tested if the samples were paired using 'Two-Related-Sample' test, that is if there were any differences in the mean or not. Statistical significance was set at $p < 0.05$ for all tests discussed below.

7.1. Time

Time was found not to follow a normal distribution and samples were not paired. For this reason, we used the Kruskal-Wallis test for comparing the result of the three techniques. The result showed no significant difference between the three techniques with respect to the time taken to answer the questions ($p = 0.7$). Still, participants spent least time, (6.1 min) on Sunburst Pop-Up compared to F+C Icicle (7.7 min) and Stacked Boxes (7.6 min) (Figure 16).

7.2. Error

The **error in the** *CS* task was found not to follow a normal distribution and did not involve paired samples. For these reasons, we applied the Kruskal-Wallis test for significance testing. The result showed a significant effect of technique ($p = 0.008$). For pairwise comparison, we employed the Mann-Whitney U test, showing better performance for Sunburst Pop-Up (5%) over F+C Icicle (26%, $p = 0.002$). We could not find any other differences (Stacked Boxes=14%) (Figure 17(a)).

Error for the C task was found not to follow a normal distribution, but it was paired samples. We applied the Friedman test. The result showed a significant effect of technique ($p = 0.004$). For pairwise comparison, we applied the Wilcoxon signed rank test between each pair of techniques and the outcomes showed that Sunburst Pop-
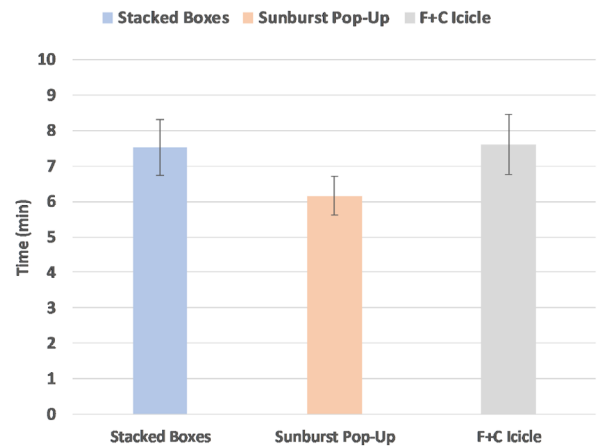

Figure 16: *The average time, in minutes, for each technique and all tasks. No significant difference between the three techniques as ($p = 0.7$).*

Up (19%) performed better than F+C Icicle (71%, $p = 0.002$). We did not find any other differences (Figure 17(b)).

For the S task, it was not a normal distribution and not paired samples. We applied the Kruskal-Wallis test and we could *not* find any significant differences, yet error was higher for F+C Icicle than the two other techniques ($p = 0.35$) (Stacked Boxes= 24%, Sunburst Pop-Up= 19%, F+C Icicle= 38%) (Figure 17(c)).

7.3. Subjective ratings

After the study, the participants were asked to provide feedback about their experience with each technique. Participants rated each of the three techniques across three satisfaction questions covering the search, reading the text, and exploring the structure of the argument using the layout. Responses were on a 1–5 Likert scale with 1 = Very easy and 5 = Very difficult. For observing search results, the Kruskal-Wallis showed no difference between the three techniques ($p = 0.09$). The results clarified that the participants found Sunburst Pop-Up and Stacked Boxes the easiest compared with F+C Icicle in finding the search results which scored the highest number in difficulty. For text reading, Friedman's ANOVA showed significant difference between techniques ($X^2 = 4.669$, $p < 0.01$). A Wilcoxon signed rank test for pairwise comparison showed that stacked pop-up labels in Sunburst Pop-Up ($p < 0.01$) and Stacked Boxes ($p = 0.001 < 0.01$) were preferred over F+C Icicle ($p < 0.01$). The results clarified that the techniques are used on Sunburst Pop-Up and Stacked Boxes were the easiest techniques compared with F+C on F+C Icicle. For overview, Friedman's ANOVA test was applied, and the results showed a significant effect of technique ($X^2 = 27.492$, $p < 0.01$). A Wilcoxon signed rank test for pairwise comparison showed that Sunburst Pop-Up was preferred over F+C Icicle and Stacked Boxes. The results illustrate that the Sunburst Pop-Up and Stacked Boxes were the most straightforward techniques compared to F+C Icicle.

We also asked our participants to rank the techniques from the most favourite to the least (Figure 18). Sunburst Pop-Up turned
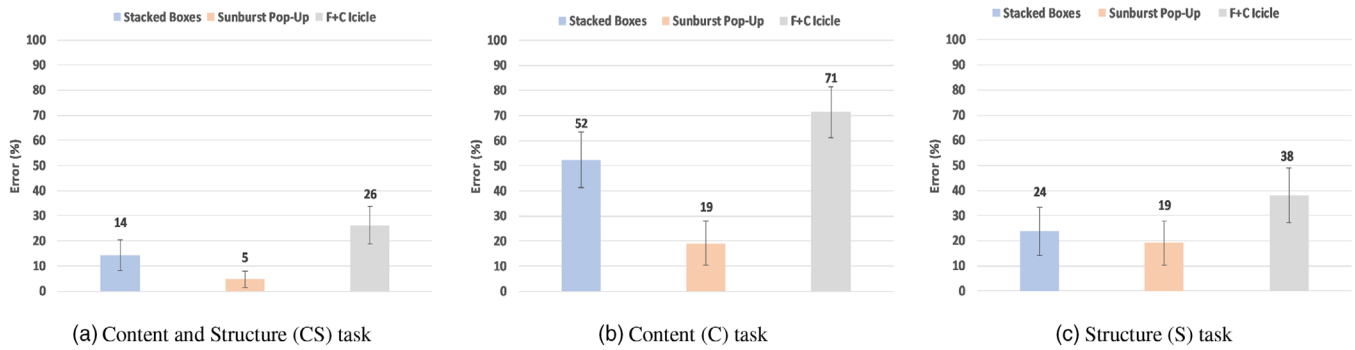
(a) Content and Structure (CS) task

(b) Content (C) task

(c) Structure (S) task

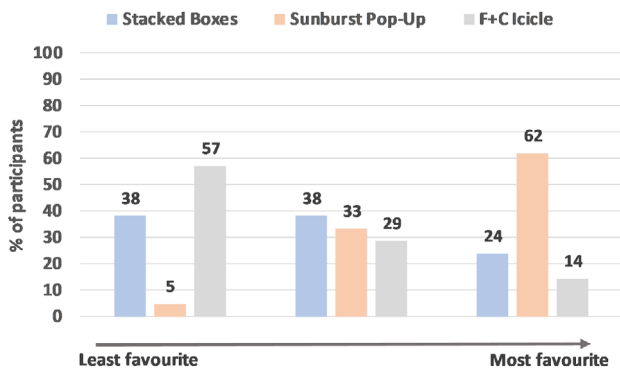**Figure 17:** *Error for each task by technique.*

Figure 18: *The percentage of the participants with their order of the favourite techniques.*

out the most favourite technique (62%). While the F+C Icicle was ranked most often as the least favourite technique (57%) and the Stacked Boxes is the second favourite technique. The participants found the sunburst layout to display the structure better than the icicle layout. In the following section, we discuss in detail the evaluation results and the hypotheses.

7.4. Discussion

Our goal with this study was to find the technique that performed best across our three tasks concerning time and error. Results suggest that there is no significant difference in time between the techniques. However, the time that the participants spent answering the questions in Sunburst Pop-Up was less than the other techniques. Also, Sunburst Pop-Up had better performance than F+C Icicle in both CS and C tasks. However, there is no difference between the three techniques in the S task.

Regarding the subjective rating, our results show that Sunburst Pop-Up is the most straightforward technique to navigate and that the participants found the stacked pop-up labels straightforward for reading text with some comments about the position of the list. This is evident in Table 1, which confirms the accuracy of the Sunburst Pop-Up being highest in all the tasks compared with the other two techniques. Besides, we can notice that the accuracy of Stacked

Table 1: *The accuracy of the three techniques in all the tasks.*

	Sunburst Pop-Up	F+C Icicle	Stacked Boxes
CS	95%	74%	86%
C	81%	29%	48%
S	81%	62%	76%
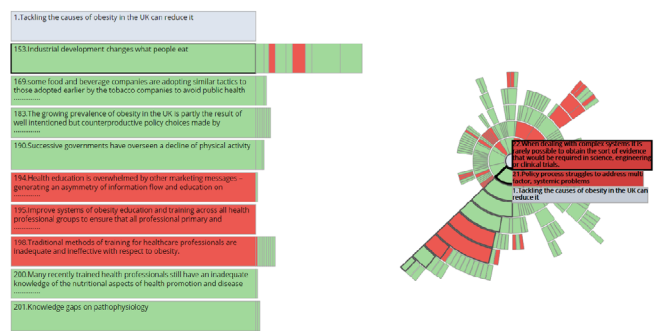
Figure 19: *The hybrid prototype between Sunburst Pop-Up and Stacked Boxes.*

Boxes is higher than the F+C Icicle. Stacked Boxes includes many features, which require more time for the participants to learn.

For our hypotheses, we can partially accept H1 for the accuracy that Sunburst Pop-Up performed better and was preferred by the participants to F+C Icicle. Stacked pop-up labels in Sunburst Pop-Up was proven better compared to the F+C in F+C Icicle, so we reject H2. We cannot accept or refute hypothesis H3 as our results do not show evidence that Stacked Boxes are more or less accurate than the other techniques in the statistical test. However, we can see in Table 1 that the Stacked Boxes performed better than the F+C Icicle.

Overall, we can say that Sunburst Pop-Up performed better than the other techniques and was more preferred by the participants. Some suggestions were made to improve the position of the stacked pop-up labels. In Section 5.3 we already presented the additional three options (**Label-1** (Figure 10e), **Label-2** (Figure 10f), **Label-3** (Figure 10g)) for the position of stacked pop-up labels, which were then discussed with the experts.

Table 2: *The advantages and disadvantages of the argument techniques from argument experts review.*

	Advantages		Disadvantages	
Stacked Boxes	E1:	Presenting the text in stacked boxes is incredibly readable.		
	E2:	The prototype provides an overview and helps to read and navigate through nodes.		
Sunburst Pop-Up	E1:	Fast navigation through the nodes in sunburst which help to read the text fast without losing the structure.	E2:	The only way to read the text is only hover over the nodes.
	E2:	The sunburst in this prototype is very attractive as it is big and shows a clear argument structure.		
F+C Icicle	E2:	Icicle layout shows the thread of the arguments.	E2:	The nodes in the icicle layout become very small when you go far away from the centre. It does not give a sense of overview.
	E3:	The text in the icicle layout is more readable	E3:	The disadvantage of the icicle presentation is that the first level of nodes it looks more important than the one far from the centre. which may misleading to the user.
	E4:	The icicle presentation works nicely. Also, the size of the nodes in the icicle can hold the text.	E4:	The sunburst layout is easier and intuitive to understand the argument structure than the icicle layout.
Hybrid	E3:	The combination of Stacked boxes and Sunburst Pop-up gives more option for reading the text and navigation.		
	E4:	The combination makes a lot of sense for the people who are working in argumentation. It gives us the benefit of seeing the overview structure merge with the benefit of reading and navigate the text. The sunburst is easy to engage and easy to see the depth and density of the arguments.		

8. Expert Feedback

We interviewed four experts (E1–E4) to evaluate the three techniques from the argumentation perspective. The experts work on different argument domains, including argument visualization, argument mining and sustainable travel behaviours which aims to use the arguments and reasoning to change personal transportation habits, for example, encouraging cycle use or discouraging car use. Three of the experts were involved in our user requirements collection interviews. Each one-to-one interview lasted between 60 and 90 minutes. In the evaluation session, we went through the previous three prototypes. The experts tried the three techniques and conversed about the proposed alternative three positions of the stacked pop-up labels, that is Figure 10a–c. After that, we discussed the advantages and disadvantages of each technique.

The interviews with the first two experts showed that they were in favour of Stacked Boxes. They found that Stacked Boxes offered a fixed framework to read the text and navigate through the nodes.

Based on this feedback, we decided to combine Stacked Boxes and Sunburst Pop-Up into a new hybrid design, shown in Figure 19. It displays one argument element chain in the Stacked Boxes and allows users to navigate through the sunburst layout to explore and read more argument elements using the stacked pop-up labels. This combination is especially helpful if the user wants to compare two argument element chains. In this case, the text of one argument can be displayed in the Stacked Boxes, and users can go through the sunburst layout and read the text of the other argument elements using the stacked pop-up labels. The combination satisfies user requirement (R4).

After implementing the combined prototype, we continued our interviews with the other two experts. During these interviews, we presented all four prototypes. The experts agreed with the previous two that the Stacked Boxes technique helps in reading and navigating through the text. They also acknowledged the benefits of the combined prototype, especially in terms of the navigation features offered. Some feedback from the experts is illustrated in Table 2.

The positions of the stacked pop-up labels were discussed with the experts. The result showed that **Label-2** (Figure 10f) is preferred as it is near the focused node and shows the same order of the argument in the graph. Both expert and non-expert participants agreed that the sunburst layout presents an overview of the argument data better than the icicle layout. However, the text in the first tiers of the icicle layout can be easily read due to the large size of the nodes.

While non-expert participants in our user study found that the Sunburst Pop-Up is fast and easy, experts showed a preference for Stacked Boxes due to their need to read the text carefully and understand it to analyse the structure of the argument, which they can do using the Stacked Boxes. Meanwhile, the separate radial layout is useful to show the overall structure of the argument.

9. Conclusions and Future Work

In this paper, four techniques were proposed to help experts to visualize, explore, and understand arguments. The outcomes are promising but there is still room for further improvement. First, three real datasets from DebateGraph were used to compare and evaluate the proposed techniques. The size of the datasets was around 202 nodes. In future, the techniques will be deployed on larger datasets to check if the size of the dataset affects the obtained results. Second, it will be potentially more effective if the experts test the techniques using their own datasets and tasks to provide feedback about their experience. Last, the evaluation with experts was crucial for this study. However, the sample size of the experts was relatively small due to the difficulty in finding a large number of experts who were willing to allocate time for a formal evaluation. Increasing the number of participants could potentially increase the reliability of the study outcomes.

However, there are still new challenges, ideas, and problems to cover as future work. More text analysis techniques can be applied to help experts when reading and understanding the text. For example, a summary of displayed arguments can be produced to highlight the main topics in given datasets. Involving information about the participants who engage in discussion/debate and the relations between them can shed light on the nature of the deliberation.

The proposed techniques have mainly focused on tree argument datasets. To be inclusive of all argument dataset types, the research can be extended for complex network argument datasets.

Furthermore, the red and green colours, which were used to reflect the oppose and support relations, are problematic for some people with colour vision deficiency (CVD). The colour has been changed to an accessible palette on the final version of our techniques. However, the effectiveness of this approach with arguments is yet to be tested. Therefore, a colour study needs to be conducted to explore alternative colour schemes that can better reflect the support and oppose relationships between the arguments while considering colour blindness. One immediate goal is to incorporate these visualisation techniques into the existing open-source, web-based, argumentation analysis software MonkeyPuzzle [WD17] which is a part of the Open Argumentation Platform [Wel20]. It is hoped that such efforts will lead to wider visibility and uptake of these techniques within the argumentation community.

Acknowledgements

References

[Abd10] ABDI H.: Holm's sequential Bonferroni procedure. *Encyclopedia of Research Design 1*, 8 (2010), 1–8.

[And06] ANDRIESSEN J.: Arguing to learn. In *The Cambridge Handbook of: The Learning Sciences*. Cambridge University Press, 2006, pp. 443–459.

[AWK*18] AJJOUR Y., WACHSMUTH H., KIESEL D., RIEHMANN P., FAN F., CASTIGLIA G., ADEJOH R., FRÖHLICH B., STEIN B.: Visualization of the topic space of argument search results in args. me. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (Brussels, Belgium, 2018), pp. 60–65.

[BC12] BETZ G., CACEAN S.: *Ethical Aspects of Climate Engineering*. KIT Scientific Publishing, 2012.

[Bel97] BELL P.: Using argument representations to make thinking visible for individuals and groups. In *Proceedings of the International Conference on Computer Support for Collaborative Learning (CSCL97)* (Toronto, Ontario, Canada, 1997), Int. Society of the Learning Sciences, pp. 10–19.

[BKSW07] BEKOS M. A., KAUFMANN M., SYMVONIS A., WOLFF A.: Boundary labeling: models and efficient algorithms for rectangular maps. *Computational Geometry 36*, 3 (2007), 215–236.

[BM12] BENN N., MACINTOSH A.: Policy commons–visualizing arguments in policy consultation. In *International Conference on Electronic Participation* (Kristiansand, Norway, 2012), Springer, pp. 61–72.

[BNK16] BACHER I., NAMEE B., KELLEHER J.: Using Icicle Trees to Encode the Hierarchical Structure of Source Code (2016). In EuroVis 2016 - Short Papers. The Eurographics Association.

[BP08] BALDWIN P., PRICE D.: Debategraph, 2008. https://debategraph.org/.

[BPB15] BULLEN V., PRICE D., BULLEN V.: Debategraph: a new way to address the complexity of obesity. *British Journal of Obesity 1* (2015), 54–59.

[BRL09] BERTINI E., RIGAMONTI M., LALANNE D.: Extended excentric labeling. *Computer Graphics Forum* (2009), *28*, 927–934.

[CCP09] COLLINS C., CARPENDALE S., PENN G.: Docuburst: visualizing document content using language structure. *Computer Graphics Forum* (2009), *28*, 1039–1046.

[CH19] CHALAGUINE L. A., HUNTER A.: Knowledge acquisition and corpus for argumentation-based chatbots. In *CEUR Workshop Proceedings* (Como, Italy, 2019), vol. 2528, CEUR Workshop Proceedings, pp. 1–14.

[CK08] CRAIG P., KENNEDY J.: Concept relationship editor: a visual interface to support the assertion of synonymy relationships between taxonomic classifications. In *Visualization and Data Analysis 2008* (2008), International Society for Optics and Photonics.

[CM07] CAWTHON N., MOERE A. V.: The effect of aesthetic on the usability of data visualization. In *International Conference on Information Visualization, IV'07, 2007* (Zurich, Switzerland, 2007), IEEE, pp. 637–648.

[CMS99] CARD S. K., MACKINLAY J. D., SHNEIDERMAN B.: *Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999, pp. 579–581.

[FHS*12] FINK M., HAUNERT J.-H., SCHULZ A., SPOERHASE J., WOLFF A.: Algorithms for labeling focus regions. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2583–2592.

[FNS13] FU B., NOY N. F., STOREY M.-A.: Indented tree or graph? a usability study of ontology visualization techniques in the context of class mapping evaluation. In *International Semantic Web Conference* (Sydney, Australia, 2013), Springer, pp. 117–134.

[FP99] FEKETE J.-D., PLAISANT C.: Excentric labeling: dynamic neighborhood labeling for data visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh Pennsylvania USA, 1999), ACM, pp. 512–519.

[GEAHJ*17] GOLD V., EL-ASSADY M., HAUTLI-JANISZ A., BÖGEL T., ROHRDANTZ C., BUTT M., HOLZINGER K., KEIM D.: Visual linguistic analysis of political discussions: Measuring deliberative quality. *Digital Scholarship in the Humanities 32*, 1 (2017), 141–158.

[Goo00] GOODWIN J.: Wigmore's chart method. *Informal Logic 20*, 3 (2000).

[Gov13] GOVIER T.: *A Practical Study of Argument*. Cengage Learning, 2013.

[HA06] HEER J., AGRAWALA M.: Software design patterns for information visualization. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 853–860.

[HB03] HARROWER M., BREWER C. A.: Colorbrewer. org: an online tool for selecting colour schemes for maps. *The Cartographic Journal 40*, 1 (2003), 27–37.

[HC98] HORN R. E., COLLINS H.: Mapping great debates: can computers think? *Nature 396*, 6710 (1998), 426–426.

[HGAS05] HARTMANN K., GÖTZELMANN T., ALI K., STROTHOTTE T.: Metrics for functional and aesthetic label layouts. In *International Symposium on Smart Graphics (SG 2005)* (Frauenworth, Germany, 2005), Springer, pp. 115–126.

[IQDLBS16] IANDOLI L., QUINTO I., DE LIDDO A., BUCKINGHAM SHUM S.: On online collaboration and construction of shared knowledge: Assessing mediation capability in computer supported argument visualization tools. *Journal of the Association for Information Science and Technology 67*, 5 (2016), 1052–1067.

[IZCC08] ISENBERG P., ZUK T., COLLINS C., CARPENDALE S.: Grounded evaluation for information visualizations. In *Proceedings of the Conference on BEyond time and errors: novel evaLuation methods for Information Visualization (BELIV)* (Florence, Italy, 2008). https://doi.org/10.1145/1377966.1377974.

[JS91] JOHNSON B., SHNEIDERMAN B.: Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Proceeding Visualization '91* (San Diego, CA, USA, 1991), pp. 284–291. https://doi.org/10.1109/VISUAL.1991.175815.

[KBSC12] KIRSCHNER P. A., BUCKINGHAM-SHUM S. J., CARR C. S.: *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-making*. Springer Science & Business Media, 2012.

[KI08] KLEIN M., IANDOLI L.: *Supporting Collaborative Deliberation using a Large-scale Argumentation System: The MIT Collaboratorium*. Technical Report. 4691-08, MIT Sloan, 2008.

[KL83] KRUSKAL J. B., LANDWEHR J. M.: Icicle plots: better displays for hierarchical clustering. *The American Statistician 37*, 2 (1983), 162–168.

[Kle12] KLEIN M.: Enabling large-scale deliberation using attention-mediation metrics. *Computer Supported Cooperative Work (CSCW) 21*, 4-5 (2012), 449–473.

[KLP*16] KONAT B., LAWRENCE J., PARK J., BUDZYNSKA K., REED C.: A corpus of argument networks: using graph properties to analyse divisive issues. In *Proceedings Language Resources and Evaluation Conference (LREC)* (Portoroz, Slovenia, 2016).

[KP01] KARACAPILIDIS N., PAPADIAS D.: Computer supported argumentation and collaborative decision making: the HERMES system. *Information Systems 26*, 4 (2001), 259–277. https://doi.org/10.1016/S0306-4379(01)00020-5.

[KRW*21] KIESEL D., RIEHMANN P., WACHSMUTH H., STEIN B., FROEHLICH B. Visual Analysis of Argumentation in Essays. *IEEE Transactions on Visualization and Computer Graphics*, 27, 2 (2021), 1139–1148. https://doi.org/10.1109/tvcg.2020.3030425.

[LS13] LIDDO A. D., SHUM S. B.: The evidence hub: harnessing the collective intelligence of communities to build evidence-based knowledge. In *Large Scale Ideation and Deliberation Workshop* (2013). http://oro.open.ac.uk/38002/.

[MB10] MARUSTERI M., BACAREA V.: Comparing groups for statistical differences: How to choose the right statistical test? *Biochemia medica 20*, 1 (2010), 15–32.

[Mun09] MUNZNER T.: A nested process model for visualization design and validation. *IEEE Transactions on Visualization & Computer Graphics*, 6 (2009), 921–928.

[Mun14] MUNZNER T.: *Visualization Analysis and Design*. AK Peters/CRC Press, 2014.

[OJP14] OELTZE-JAFRA S., PREIM B.: Survey of labeling techniques in medical visualizations. In *Eurographics Workshop on Visual Computing for Biology and Medicine* (2014), Viola I., Buehler K., Ropinski T., (Eds.), The Eurographics Association. https://doi.org/10.2312/vcbm.20141192.

[PEAS*18] PLÜ SS B., EL-ASSADY M., SPERRLE F., GOLD V., BUDZYNSKA K., HAUTLI A., REED C.: Add-up: Visual analytics for augmented deliberative democracy. In *7th International Conference on Computational Models of Argument (COMMA)* (Warsaw, Poland, 2018). https://doi.org/10.3233/978-1-61499-906-5-471.

[Pit07] PITSOS E.: Kialo, 2007. https://www.kialo.com/.

[RHR17] ROBERTS J. C., HEADLEAND C. J., RITSOS P. D.: *Five Design-Sheets: Creative Design and Sketching for Computing and Visualisation*. Springer, 2017. https://www.springer.com/gb/book/9783319556260.

[RMRW06] ROWE G., MACAGNO F., REED C., WALTON D.: Araucaria as a tool for diagramming arguments in teaching and studying philosophy. *Teaching Philosophy 29*, 2 (2006), 111–124.

[RR04] REED C., ROWE G.: Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools 13*, 04 (2004), 961–979.

[SGP13a] SCHNEIDER J., GROZA T., PASSANT A.: A review of argumentation for the social semantic web. *Semantic Web 4*, 2 (2013), 159–218.

[SGP13b] SCHNEIDER J., GROZA T., PASSANT A.: A review of argumentation for the social semantic web. *Semant. Web 4*, 2 (2013), 159–218. http://dl.acm.org/citation.cfm?id=2590215.2590218.

[Shn92] SHNEIDERMAN B.: Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics (TOG) 11*, 1 (1992), 92–99.

[SLPM10] SCHEUER O., LOLL F., PINKWART N., MCLAREN B. M.: Computer-supported argumentation: a review of the state of the art. *International Journal of Computer-Supported Collaborative Learning 5*, 1 (2010), 43–102.

[SP06] SHNEIDERMAN B., PLAISANT C.: Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies. In *Proceedings of the 2006 AVI workshop on BEyond time and errors: Novel Evaluation Methods for Information Visualization* (Venice, Italy, 2006), ACM, pp. 1–7.

[SS19] STEDE M., SCHNEIDER J.: *Argumentation Mining*. Synthesis Lectures on Human Language Technologies. Morgan and Claypool, 2019.

[SSKEA19] Sperrle F., Sevastjanova R., Kehlbeck R., El-Assady M.: VIANA: Visual Interactive Annotation of Argumentation (2019). In Proceedings of IEEE Conference on Visual Analytics Science and Technology (VAST), 2019, pp. 11-22, https://doi.org/10.1109/VAST47406.2019.8986917.

[Sut03] SUTHERS D. D.: Representational guidance for collaborative inquiry. In *Arguing to Learn*. Springer, 2003, pp. 27–46.

[SVB07] SCHNEIDER D. C., VOIGT C., BETZ G.: Argunet-a software tool for collaborative argumentation analysis and research. In *7th Workshop on Computational Models of Natural Argument (CMNA VII)* (Hyderabad, India, 2007).

[SWCP95] SUTHERS D., WEINER A., CONNELLY J., PAOLUCCI M.: Belvedere: engaging students in critical discussion of science and public policy issues. In *Proceedings of the World Conference on Artificial Intelligence in Education* (Washington, DC, 1995), pp. 266–273.

[SZ00] STASKO J., ZHANG E.: Focus+ context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *IEEE Symposium on Information Visualization* (Salt Lake City, UT, USA, 2000), IEEE, pp. 57–65.

[Tou58] TOULMIN S.: *The Uses of Argument*. Cambridge University Press, 1958.

[UDLB19] ULLMANN T. D., DE LIDDO A., BACHLER M.: A visualisation dashboard for contested collective intelligence. Learning analytics to improve sensemaking of group discussion. *RIED: Revista Iboeroamericana de Educación a Distancia (The Ibero-American Journal of Digital Education) 22*, 1 (2019), 41–80.

[van07] VAN GELDER T.: The rationale for Rationale. *Law, Probability and Risk 6, 1–4* (March 2007), 23–42. https://doi.org/10.1093/lpr/mgm032.

[VDBVP07] VAN DEN BRAAK S. W., VREESWIJK G. A., PRAKKEN H.: Avers: An argument visualization tool for representing stories about evidence. In *Proceedings of the International Conference on Artificial Intelligence and Law (ICAIL07)* (Stanford, California, 2007), ACM, pp. 11–15.

[vEGK*14] VAN EEMEREN F. H., GARSSEN B., KRABBE E. C. W., SNOECK HENKEMANS A. F., VERHEIJ B., WAGEMANS J. H. M.: *Handbook of Argumentation Theory*. Springer, Dordrechtx, 2014.

[Ver03] VERHEIJ B.: Artificial argument assistants for defeasible argumentation. *Artificial Intelligence 150, 1-2* (2003), 291–324.

[vG03] VAN GELDER T.: Enhancing deliberation through computer supported argument visualization. In *Visualizing Argumentation*. Springer-Verlag, Berlin, Heidelberg, 2003, pp. 97–115. http://dl.acm.org/citation.cfm?id=779331.779337.

[Wal05] WALTON D.: *Fundamentals of Critical Argumentation*. Cambridge University Press, 2005.

[War12] WARE C.: *Information Visualization: Perception for Design*. Elsevier, 2012.

[WD17] WELLS S., DOUGLAS J.: Monkeypuzzle: towards next generation, free & open-source, argument analysis tools. In *Proceedings International Workshop on Computational Models of Natural Argument (CMNA17)* (London, UK, 2017).

[Wel20] Wells S. The Open Argumentation PLatform (OAPL). Frontiers in Artificial Intelligence and Applications (2020), pp. 475–476 https://doi.org/10.3233/FAIA200541.

[WYM19] WOODBURN L., YANG Y., MARRIOTT K.: Interactive visualisation of hierarchical quantitative data: an evaluation. 96–100. https://doi.org/10.1109/VISUAL.2019.8933545.

[YFDH01] YEE K.-P., FISHER D., DHAMIJA R., HEARST M.: Animated exploration of dynamic graphs with radial layout. In *IEEE Symposium on Information Visualization, 2001. INFOVIS 2001.* (San Diego, CA, USA, 2001), pp. 43–50. https://doi.org/10.1109/INFVIS.2001.963279.

Appendix A

Table A1: *A summary of information relating to the study participants.*

	Country	City	Current Job Description	Experience
E1	**UK**	**Dundee**	He works as a research assistant in argumentation since 2009. He is working now in argument mining. He has developed a range of widely used tools for argument analysis, visualization, and storage.	10 years
E2	**UK**	**Swansea**	He is an Associate Professor in Law and Computer Science Legal Studies and works in argument mining.	10 years
E3	**USA**	**Illinois**	She is an Assistant Professor in argumentation and evidence. She is developing Linked Data (ontologies, metadata, Semantic Web) approaches to managing scientific evidence.	6 years
E4	**UK**	**Leeds**	He is a research assistant. He works in the areas of analytic aesthetics and moral psychology.	3 years
E5	**UK**	**Leeds**	She is an academic fellow. Her research interests are transporting governance, sustainable travel behaviours, persuasive technology, and transport geography.	4 years
E6	**UK**	**Edinburgh**	He worked as a retired researcher in philosophy. He worked in the project about investigated the relevance of argument visualization techniques to political debates.	4 years
E7	**UK**	**London**	She is a research associate. She is interested in network-centric, modelling and visualising ideas and arguments as networks of nodes which can be analysed for topographical and semantic patterns.	13 years

Table A2: *Interviews questions, divided into three parts; uses of argument visualization, size of the data and understanding the arguments.*

	Questions
Use of argument visualization	Q1- Do you use any argument visualization tools?
	Q2- What kind of things are you looking for by using this tool ?
	Q3- What techniques do you use to present the arguments ?
	Q4- Are there any limitations in this tool?
	Q5- What are the actions that you want to do with your data but you cannot by using this tool ?
	Q6- Can you see all the information you want? If not, what is missing?
	Q7- Why are these things (in the previous question) essential for argumentation?
Data size and obstacles	Q8- What is the size of the dataset that you have used ?
	Q9- Did you find it difficult to present this data? Why?
	Q10- How do you break down a large argument to understand it?
	Q11- What information do you like to present or see on the argument?
Reading and understanding the argument	Q12- Is there a specific flow that you want to read the arguments through ?
	Q13- When you explore the arguments, what are the things you are looking for?
	Q14- Is it important to you to know the number of nodes that support/attack the main conclusion? Or what is information that you like to know about this diagram?
	Q15- What are the features that you would like the tool to provide to help you understand the argument? Querying? Exploring? Searching?
	Q16- Why is it important to check the overview of the whole data before starting to explore one argument?
	Q17- How is it important to present the different type of arguments (linked/convergent)?

Table A3: *Questions about each data set, related to the tasks CS, C, and S for the different datasets.*

Dataset	Questions	CS	C	S
Obesity	Please provide any two statements which detail actions that help to tackle obesity and reduce it.	X		
	Which statement explains why some companies use fructose instead of cane sugar?	X		
	Which statements mention the term eating?		X	
	Which statement has the most immediate supporting statements (without opposing statements)?			X
Climate Change	Please provide two statements which explain the role of fossil fuels in climate change.	X		
	Please provide any two statements which mention the relation between climate change and water vapour.	X		
	Which statements mention the term energy?		X	
	Which statement has the most immediate opposing statements (without supporting statements)?			X
Newspapers	Please provide any two statements which explain how newspapers should enforce their copyright.	X		
	Please provide one statement which mentions who provides the best coverage of any newsworthy event.	X		
	Which statements mention the term source?		X	
	Which statement has the longest argument chain (excluding the parent node)?			X

Table A4: *The three 5-point Likert scale subjective usability questions about participants' experience on using the relevant technique.*

Tool	Questions	Reading	Search	Layout
StackedBoxes	How did you find the pop-up list of labelling when browsing for reading the text?	X		
	How easy was it to see the result of a search on the argument map?		X	
	How easy was it to explore the structure of the argument map?			X
Sunburst Pop-Up	How did you find the pop-up list when browsing for reading the text?	X		
	How easy was it to see the result of a search on the argument map?		X	
	How easy was it to explore the structure of the argument map?			X
Icicle F+C	How did you find the F+C functionality of the Icicle layout?	X		
	How easy was it to see the result of a search on the argument map?		X	
	How easy was it to explore the structure of the argument map?			X

Table A5: *Some comments provided by the participants about their favourite layouts are listed below.*

Participants	Comments
P1	I like the Sunburst Pop-Up technique as it is straightforward, easy to use and understandable
P2	When I used the Sunburst Pop-Up, I felt comfortable to answer the questions as it is easy to spot the structure of the arguments. In Stacked Boxes clicking on any nodes on the sunburst and see how the arguments flow on another view is good, but this technique took some time for me to get used of it. For me, the F+C Icicle is not tidily presented and I feel it is a messy picture.
P3	Regarding the F+C Icicle, I did not like the presentation. Finding the results of the search is difficult. I liked the Stacked Boxes; the navigation using the keyboard is a nice feature, especially using Alt+S to navigate through the search results. I did not use the sunburst in the coordinated view at all until I came through the question that related to the structure. I loved Sunburst Pop-Up as it is very easy to navigate and read the text and it is my favourite.'
P4	I hate the F+C Icicle. It is not familiar to the eyes. The size of the nodes is very different, and some nodes are tiny. The Sunburst Pop-Up is very easy; the structure is clear. Nothing complicated in it.