Enhancing Text Editors with Graph Visualisations

Adam Binks
University of St Andrews
14/04/20

Supervised by
Alice Toniolo
University of St Andrews

Miguel Nacenta
University of Victoria
Abstract

Two key subprocesses in academic writing are planning, in which the argument is mapped out, and writing, in which the graphical, branching reasoning structure is flattened into a linear textual format. Tools exist to aid each sub-process individually. However, as writing progresses, plans are often extended and modified. Therefore, if using separate tools for planning and writing, then as the document evolves, one or other component will lose its status as a faithful representation of the other. The author must then perform tiresome work keeping the two documents in sync, or else abandon the plan entirely as it becomes out of date, losing its valuable facility of visualising the entire reasoning structure at once.

To address this, this report presents Write Reason, a tool to unify the planning and writing sub-processes. The tool provides a novel interface combining a graph, to map reasoning structure, and a text editor, to linearise the document in textual form. This report also presents the findings of a user study comparing the quality of arguments constructed using Write Reason with those constructed using an ordinary text editor, and the kind of graphs generated by users of Write Reason.

Declaration

I declare that the material submitted for assessment is my own work except where credit is explicitly given to others by citation or acknowledgement. This work was performed during the current academic year except where otherwise stated.

The main text of this project report is 14,430 words long, including project specification and plan.

In submitting this project report to the University of St Andrews, I give permission for it to be made available for use in accordance with the regulations of the University Library. I also give permission for the title and abstract to be published and for copies of the report to be made and supplied at cost to any bona fide library or research worker, and to be made available on the World Wide Web. I retain the copyright in this work.
Contents

1 Introduction ................................................. 4
   1.1 Motivation ........................................... 4
   1.2 Overview ............................................ 4
   1.3 Requirements specification ......................... 5
       1.3.1 Original requirements ....................... 5
       1.3.2 Change of requirements ..................... 6
       1.3.3 Updated requirements ....................... 7

2 Context survey ............................................. 7
   2.1 Mindmapping ......................................... 7
   2.2 Argument mapping ................................... 8
   2.3 Text markup tools .................................. 9
   2.4 Novel contributions ................................ 9

3 Software engineering process .......................... 11
   3.1 Feature selection and prioritisation .............. 11
   3.2 Time management .................................. 11

4 Ethics ....................................................... 12

5 Design ....................................................... 13
   5.1 Design process ..................................... 13
       5.1.1 Idea generation .............................. 13
       5.1.2 Storyboarding ................................ 16
       5.1.3 Paper prototype ............................ 19
       5.1.4 Paper walkthrough .......................... 19
   5.2 Final design ......................................... 19
       5.2.1 Overview ..................................... 19
       5.2.2 Graph pane ................................... 20
       5.2.3 Document pane and interaction between panes .... 22
       5.2.4 Plain condition .............................. 24

6 Implementation ........................................... 26
   6.1 Architectural decisions ............................. 26
   6.2 Difficult implementation details .................. 28
   6.3 Testing ................................................ 29

7 Study method ............................................. 30
   7.1 Topic selection ..................................... 31
   7.2 Fact sheets .......................................... 32
   7.3 Participants and conditions ....................... 33
   7.4 Experimental procedure ........................... 34
   7.5 Apparatus .......................................... 36
   7.6 Marking .............................................. 37
8 Analysis

8.1 Difference in scores between tools ........................................... 38
8.2 Score difference between participant groups .......................... 40
8.3 Testing assumptions of t-tests ....................................................... 41
8.4 Tool preference ........................................................................ 41
8.5 Graphs constructed ..................................................................... 43
8.6 Analysis of scores ..................................................................... 48
8.7 Use of paper .............................................................................. 50
8.8 Analysis methodology ............................................................... 50

9 Discussion and limitations .............................................................. 50

9.1 Limitations and future work ......................................................... 52

10 Evaluation .................................................................................. 53

11 Conclusions ................................................................................ 54

12 References .................................................................................. 56

Appendices ....................................................................................... 58

A User manual ................................................................................ 58

B Fact sheets used in study ............................................................... 58

C Ethical approval ............................................................................ 63
1 Introduction

1.1 Motivation

In academic writing, a core objective is the construction of a clear, well supported reasoning structure, transmitting confidence in reasons and empirical evidence to confidence in conclusions, by means of argumentation.

Two key subprocesses in academic writing are planning, in which the parts of the argument, and the relationships between them, are mapped out, and writing, in which the graphical, potentially branching, reasoning structure is flattened into a linear textual format. When planning, visual techniques such as mindmapping are used to represent the complex, branching nature of the reasoning landscape in a more natural way than free text.

Existing text editors do not support the interplay and co-evolution of these two processes. As the writing process takes place, and elements of the plan are expanded upon as sections of the text, the plan is often extended and modified. Thus, if two separate mediums or tools are used to do planning and writing, as the document evolves, one or other component will lose its status as a faithful representation of the other. The author must then perform tiresome work keeping the two documents in sync, or else abandon the plan as it becomes out of date, losing its valuable facility of visualising the entire reasoning structure at once.

The central research question of this project is: how does graph-based visualisation of reasoning structure affect argument construction in academic writing? To that end, this project concerns the design and implementation of an application to unify the two co-dependent processes of planning and writing. The application is then evaluated by means of a user study, comparing the quality and kind of arguments generated with the novel application to those generated using a typical text editor.

Finding ways to improve the quality and validity of academic writing has the potential for a large positive impact - if these findings are implemented in essay writing software, then the quality of research generation and communication could be improved, leading to more accurate beliefs amongst the academic community and the application of accurate research findings in high impact domains.

1.2 Overview

This report first details the requirements, design and implementation of a novel tool for planning and writing in an academic context: “Write Reason”. Our survey described in Subsection 2.4 finds that Write Reason is unique amongst tools as it provides a unified interface for both writing in a linear textual format using a text editor, and building an argument map using a graphical interface.
This report then describes the methods and results of a user study conducted to evaluate the tool. In the study, 24 university students used both Write Reason and a plain text editor to construct arguments, which were scored by expert markers. The study found that arguments constructed using the plain text editor scored significantly higher on one score, persuasiveness, than those constructed using Write Reason. No significant effect was found on other scores. These results are analysed, motivating further studies to explore this unexpected result. Additionally, the kinds of graphs generated using Write Reason are analysed, presenting a novel contribution by observing which strategies are chosen when participants freely choose how to best represent their ideas using a graph.

Figure 1: A screenshot of Write Reason

1.3 Requirements specification

1.3.1 Original requirements

The initial requirements, as decided at the start of the project, are described below:

Primary

(P1-orig) Design and build an application which combines text editing facilities with tools to plan, communicate and recall a graph of the reasoning structure

(P2-orig) The application supports “from scratch” authoring of projects by planning the structure of the piece and subsequently filling the prose, while maintaining and updating the structural map
(P3-orig) The application supports mark-up of existing pieces of academic writing, to extract structure and aid understanding and critical analysis

Secondary

(S1-orig) The application supports real-time collaboration between different users in the same project

(S2-orig) The application supports templating for graph node types

(S3-orig) The application supports templating for graph structures

Tertiary

(T1-orig) Conduct a user study to determine the effect of the application on the collaborative and individual ideation and writing processes

(T2-orig) Release the tool to the world to see if it’s picked up by users

1.3.2 Change of requirements

The project underwent a change of focus during the early implementation phase, as is typical in a highly research-based project such as this one. Initially, the primary focus was on the development of a fully featured prototype of the Write Reason tool, incorporating non-essential desirable features such as (S1-orig), (S2-orig) and (S3-orig). However, we decided that a more substantial contribution to the literature, that better fulfils the initial motivation of the project, would be a robust user study, to determine the effects of a graph-based visualisation on argument construction.

This more ambitious goal required focusing more of the project’s effort on the design, execution and analysis of the user study. As such, the Write Reason prototype’s development needed to be completed much earlier. Further, a more stripped back feature set would allow a clearer analysis of whether the addition of the graph in particular contributed to a significant difference in the kind and quality of arguments constructed - the presence of additional features would make it harder to distinguish whether any observed effects were due to these additional features or the graph-based interface itself.

With this in mind, the initial requirements specification was no longer fit for purpose - (T1-orig), a tertiary requirement, was now the core primary goal of the project. Meanwhile, the secondary goals (S1-orig), (S2-orig) and (S3-orig), as well as (P3-orig) - which fell beyond the scope of the user study - were no longer important requirements, and were deprioritised accordingly. These changes are reflected below in a reprioritised list of requirements.
1.3.3 Updated requirements

Primary

(P1) Design and build an application which combines text editing facilities with tools to plan, communicate and recall a graph of the reasoning structure

(P2) The application supports “from scratch” authoring of projects by planning the structure of the piece and subsequently filling the prose, while maintaining and updating the structural map

(P3) Conduct a user study to determine the effect of the application on the individual ideation and writing process

Secondary

(S1) The application supports mark-up of existing pieces of academic writing, to extract structure and aid understanding and critical analysis

Tertiary

(T1) The application supports real-time collaboration between different users in the same project

(T2) The application supports templating for graph node types

(T3) The application supports templating for graph structures

(T4) Release the tool to the world to see if it’s picked up by users

2 Context survey

In this survey I discuss existing tools and their study. Write Reason provides a flexible interface which can be used for either mindmapping or argument mapping. Thus, I focus on the existing tools for these two techniques, and their study, as this is the broader literature to which Write Reason contributes. I then discuss the novel contributions of this project.

2.1 Mindmapping

Mindmapping, originally coined by Osborn [10] in 1953, has gained widespread popular usage, reflected by the number of commercially available tools for the creation of mind maps. Products typically allow the creation and labelling of nodes, with varying degrees of structure imposed. Some tools, such as Coggle
(coggle.it) and MindMup (mindmup.com) enforce a radial tree structure where nodes are equally spaced around their parent node. Others, such as Scapple (literatureandlatte.com/scapple) allow nodes to be placed freely on the plane, and also allow disconnected subgraphs rather than requiring all nodes to be descendants of an initial central node. Real-time collaborative mindmapping tools, such as Mindmeister (mindmeister.com) are also available, allowing a Google Docs-style multi-user approach to mindmapping. While previously mentioned tools are for web or desktop, mobile-first solutions have also been developed, such as Mindly (mindlyapp.com).

The effect of mindmapping in the educational context is much studied. Linder et al [7] explored the effect of tools in the mindmapping paradigm for presenting information in the classroom, finding that free-form presentation encourages free-form thinking, spontaneous discussion and emergent ideation. Al-Jarf et al [2] found that students who used mind-mapping for essay writing made greater improvements in their writing than a control group. In [1], Adodo found that secondary school science students improved their Basic Science and Technology scores to a greater extent if they used mindmapping than a control group.

Outside of the educational context, a collaborative mindmapping tool, GroupMind, was developed and studied by Shih et al in [13], and performed better than a traditional whiteboard for group tasks involving memory recall. Relatedly, Carneiro et al [3] present Deb8, a collaborative fact checking tool which adopts a mindmapping-esque interface for analysis of video debates.

2.2 Argument mapping

Mindmapping is a graph structure in which nodes are connected if they relate to one another, typically oriented around a central topic node. On the other hand, argument mapping or argument diagramming is the visual representation of the structure of an argument. There are many systems of argument mapping, some of which are compared by Reed et al in [12]. Argument maps typically represent an argument’s premises and conclusions, showing the relevant claims and the relationship between them.

In the educational context, Harrell’s study [4] of the effects of teaching argument diagramming to students in an introductory university philosophy course found that low- and mid-achieving students’ scores significantly increased when taught argument diagramming techniques, though high-achieving students’ scores were not affected. Harrell et al’s later study [5] found that visual representations of argument structure had a significant positive effect on the outcomes of students in a first year college writing course.

Various commercial and academic software tools for argument mapping have been developed, such as Rationale (reasoninglab.com/rationale), bCisive (reasoninglab.com/bcisive), Compendium [15], AGORA-net [6] and OVA-net [9].

The methodology and validity of studies on tools developed in prior work in the
area has been critically reviewed by van den Braak et al in [16], finding that earlier work in the area often lacked internal validity and thus limited conclusions can be drawn.

Argument mapping tools enforce different mapping systems to different degrees of stricture. In line with results found by Shipman et al in [14] that formal structures can impede creativity, a design goal of the novel tool presented in this paper, Write Reason, was to allow users to implement whichever diagramming system that came most naturally to them. This marks this work out as a novel contribution, as the fact that prior tools enforce a specific argument mapping technique means there is no opportunity to study what kinds of techniques users select when allowed to choose their own freely.

2.3 Text markup tools

The previous sections on mindmapping and argument mapping relate to Write Reason’s graph pane. This section focuses on prior work relating to the Write Reason’s text editor. Text markup tools allow users to write or load a corpus of text, then assign attributes to words and phrases within it. A common application of such tools is in text annotation in Natural Language Processing and linguistics.

One approach to text markup is text segmentation, as used by the Systemic Coder (wagsoft.com/Coder) RST Tool (wagsoft.com/RSTTool), where RST means Rhetorical Structure Theory [8]. This method allows users to mark the boundaries between segments in a text, by adding segment dividers. However, a constraint of text segmentation is that all text is initially part of a single segment, and this can only be subdivided - every word must be part of a segment. Other tools dispense with this constraint, such as Dexter Coder (dextercoder.org) which allows visual annotation of text by highlighting selected strings in colours based on the code being assigned. This interaction pattern is used by Write Reason in its document pane. The UAM Corpus-Tool (http://corpustool.com/features.html) also allows users to markup strings by highlighting them, and subsequently allows properties of a annotated word or phrase to be assigned. Another, more specific, text markup approach is that of Simple PoS Tagger (martinweisser.org/lingsoft.html#tagger), where PoS denotes “Part of Sentence”. In this tool, each word in the text can be assigned a tag, such as “Noun (general, plural)”, rendering it useful for its specific application in linguistics.

2.4 Novel contributions

This survey of existing tools and literature has found that no tool exists which combines an argument mapping interface with a text editor for essay authoring and argument construction. All of the tools described in Subsections 2.1 and 2.2
implement either a mindmapping or argument mapping interface, but with no continuous connection to an interface for writing the argument down in linear textual form.

The mindmapping tool WriteMapper (writemapper.com) allows one-time exporting of a mind map to a Word document or similar formats, by automatically flattening the mind map into a document by interpreting it as an ordered tree, where child nodes positioned directly above the parent node are inserted first, then subsequent nodes are inserted in a clockwise direction. However, this does not achieve the goal of this project, which is to maintain a continuous synchronised state of the plan, in the form of the graph, and the essay, in the form of the document - in WriteMapper, once the Word document is generated, if the mind map or the document are edited, these changes are not reflected in the other representation.

The only tool found which combines an argument mapping or mindmapping approach with an interface for essay writing is Essaywriter (essaywriter.com), a commercial tool originally developed for students with dyslexia, mild autism and aspergers, to assist learning. Essaywriter has a split pane interface in which one pane allows users to build a mind map of their ideas, and the other pane is a text editor for the authoring of the essay. Nodes in the mind map are inserted into the essay based on their ordering, in a similar system to WriteMapper as described above.

Key differences distinguish Write Reason, the novel tool presented in this paper, from Essaywriter. Essaywriter allows users to build a mindmap, in which nodes are all descendants of one central topic node, and relations are strictly tree-like (nodes all have exactly one parent and can have multiple descendants). Write Reason allows users to build a flexible argument map, in which nodes can be connected freely - multiple connections can be made between any nodes, and nodes can also be totally disconnected. Additionally, unlike in Essaywriter, connections in Write Reason can be of different types, as appropriate for an argument map: “supports”, “opposes” and “expands on”. A further difference is that, in Essaywriter, all nodes are inserted into the document upon creation, and the ordering is totally determined by the placement of nodes on the graph - the node positioned directly above a parent node is inserted first, then subsequent children are inserted in clockwise ordering. In Write Reason, not all nodes need to be represented in the graph, allowing the graph to be used for a full mapping of the argument landscape, including claims, objections and replies which the user can decide not to include in the essay. Additionally, nodes in Write Reason can be freely positioned to allow the user to spatially organise their ideas.

The further novel contribution of this work is the experimental evaluation: Write Reason’s effects on argument quality and kind, as well as the kinds of graph structures generated using the tool, have been the subject of a user study.
3 Software engineering process

3.1 Feature selection and prioritisation

The approach taken to software engineering was strongly results oriented - the goal of creating the prototype tool was to provide the basis for the user study. In the design phase (described in detail in section 5), a long list of potential features was generated, each of which we anticipated furthering the power or usability of the tool.

After this creative process of ideation, a more critically-focused phase of prioritisation was undertaken, in which the features were categorised using Trello (trello.com) according to how critical they were in generating insightful and interpretable results from the user study. Features were first categorised as essential, desirable, or 'nice to have'. Then, a second pass was made, critically re-examining the initial taxonomy, and moving the most important 'essential' features into the category of being required for the Minimum Viable Product (MVP). Within the MVP list, features were prioritised in order of importance for the end product, and for the architecture of the implementation - capturing the reality that, while some features do not directly impact on the user experience and so were not directly required for a successful user study, they were architectural requirements for features which were directly required, and so were instrumentally highly important.

Once the features were ordered, a final pass was made to decompose the features into granular implementational tasks. This allowed easier prediction of the amount of work required to complete each task, and provided a preview of upcoming tasks, to identify useful common abstractions which would save time and improve the maintainability of the codebase.

The task of deciding which feature to implement next was made trivial, as I simply needed to pop the next feature off the top of the prioritised queue. This was a useful software engineering methodology, as it required less context switching between the design and implementational mindsets. Instead big picture prioritisation work was all done at once. Then, once I was in the depths of technical implementation work, the friction for beginning work on the next feature was minimal, promoting a state of flow and highly motivated development.

3.2 Time management

I built a Gantt chart to work backwards from the final project completion date, allocating sufficient time to design and run the user study, and analyse its results - thus determining the date at which the final software version was to be completed. From this, I was then able to divide up the design and implementation time, marking in the milestones to be reached if development was to proceed at a uniform rate and be completed in time for the next phases of the project.
This was highly useful in maintaining a sense of where the project stood, and what was coming next, providing grounding to decisions about when to apply myself in pushing for development milestones.

Additionally, I met with my supervisors on a weekly basis, to review added features, flag issues or unanticipated complications, discuss newly uncovered design considerations, and set goals for the next weekly meeting. This agile development approach helped me keep the project on a consistent upward trajectory throughout its eight month lifespan, with the regular meetings helping me avoid becoming mired in inconsequential technical problems or misallocating my efforts.

4 Ethics

Ethical approval was gained for the running of the study (see the approval document in Appendix C).

A full description and justification of the study method is given below in section 7. Ethical ramifications were considered and mitigated wherever possible. For example, participants were asked to construct arguments about two topics - these topics were selected to be areas which do not evoke mainstream controversy or have any connection to potential trauma. Rather, they were neutral areas which participants are unlikely to have strong opinions about.

Participants were informed that their anonymised data would be shared with markers, who would assign scores based on attributes such as clarity, persuasiveness and structure. An ethical risk for participants was a potential feeling of distress due to the marking of their work. Care was taken to mitigate this risk by ensuring that participants knew that this was purely for the purpose of evaluating the kinds and quality of arguments constructed with the tools, and not to evaluate the participants themselves - in particular, that this would not affect their academic assessment in any way.

We collected screen recordings and logs from the usage of the software tools on a specially prepared computer, and questionnaire responses regarding basic demographic information, as well as participants’ experiences with the tools and prior essay writing experiences. Participants were given a £5 Amazon voucher to thank them for their time, and markers were given a £70 Amazon voucher for their more time-intensive contribution.

All data recorded was stored in fully anonymised form, and only handled by me and my supervisors (and, in the case of the anonymised arguments, the markers). Participants were informed that they could withdraw from the study at any time if they wished to do so. The risks involved in this study did not exceed those that participants would encounter during an average day.
5 Design

For the results of a user study to deliver insights about the central research topic of how graph-based visualisation affects kind and quality of arguments constructed, the tool which implemented the graph-based visualisation needed to be a useful implementation of such an idea.

Hypothesis: A graph-based visualisation of the reasoning structure aids argument construction.

This hypothesis could only be reliably disconfirmed if said visualisation was part of a genuinely usable tool with a useful and relevant feature set. Yet the precise design of this tool was not specified by the proposal. As a result, a key phase of the project was the selection of a design fulfilling these criteria, to provide a basis for insightful and interpretable results from the user study.

5.1 Design process

The design process used was an iterative process based on sketching, feature evolution and expert critique.

![Diagram of design process]

Figure 2: A summary of the design process

5.1.1 Idea generation

The first stage of the design process was the mass generation of ideas. A risk in interface design is to fixate on one of the first designs that is considered, thus ending the exploration of the larger design space prematurely - before reaching confidence that that design is indeed the one which best satisfies the design goals.

To avoid this, I generated a large number of sketches, each exemplifying a different vision of the possible user interface. This was a new approach for me, but one which was both highly enjoyable and successful in mapping a larger part of the possibility space, allowing the ideas taken forward to be composed of the strongest elements of multiple initially considered possibilities.

In particular, I aimed to generate ideas which were radical re-imaginings of how the tool could be oriented. I focused on having a generative approach rather than a critical approach, creating the mental space for more novel ideas rather
than conforming to existing design patterns by default - those common design patterns which I did eventually converge on were thus shown to be genuinely successful in achieving the design goals, rather than just the most obvious interpretations.

I initially considered different methods of representing reasoning structures than a graph, such as the idea in Figure 3 inspired by Scratch (scratch.mit.edu), a visual programming application. Users could drag logical operator blocks from a toolbox, and type claims into the “proposition” blocks. By composing these blocks, a progressively more advanced argument could be captured. Additionally, static analysis of the argument entered could alert the user to validity, and suggest fixes for fallacious reasoning. Common argumentative structures, such as modus ponens or reductio ad absurdom, could be saved as preset templates of collections of blocks for easy addition.

I moved towards a graph-based representation for greater flexibility of use, as I anticipated that it would better represent a wider set of arguments and reasoning structures without excessive detail.

In Figure 4 I considered ways in which arguments with premises and conclusions could be represented. The top option allows special nodes with premises and conclusions inside them, and the ability to draw arrows to and from a premise or conclusion by dragging from its number or the “C” for conclusion. The second option allows nodes to be clustered as arguments, and arrows can be drawn from individual nodes within a cluster or from the entire cluster itself, to allow, for example, objections to individual premises, or to the entire argument as a whole. The third option adds a special “argument” object, which is not a node, but by connecting nodes to the various circles within it lets the user mark out existing nodes as premises and conclusions in arguments. In the end, I favoured the second option for the final design. The clustering feature was
not implemented for the study version of Write Reason, but it may be a future addition to the application.

In Figure 5 I allow the user to represent their credence in a given node or arrow by dragging a bubble, in which numeric values can be entered, onto the element in question. The idea here was to promote good epistemic practise, perhaps using Bayesian methods to analyse graphs and determine whether the credences entered are rational. This is an interesting feature but goes beyond the scope of the minimal implementation for the study (it would be difficult to distinguish the effect of the graph and the credence bubbles upon argument quality).

In Figure 6 I consider a more opinionated ontology, where nodes would fall into various categories (argument, motivation, objection, response, modification, comparison, etc). In the end, I favoured the more stripped back ontology described in the final design section 5.2.2. I moved in this direction for wider applicability to a broader set of domains and personal preferences, and so that the
preferred approach of individual participants could be studied by not enforcing a structure upon them.

5.1.2 Storyboarding

Having generated a map of some interesting terrain in the design landscape, in the next stage I took the ideas from various designs which, together, best satisfied the central objective of allowing the user to construct a graph of the reasoning structure and the essay in tandem. For this goal, it was important that the design allowed the users to maintain a strong correspondence between elements in both views, and allowed their synchronisation and co-evolution. Ideas taken forward from the idea generation stage constituted most of the final design - for example, flexible un-opinionated generic nodes, arrows with types, a two pane resizeable view, and dragging content from one pane to the other.

I constructed storyboards for the two primary scenarios:

- Planning and writing an essay from scratch
- Marking up an existing essay
Representing the interface on paper forced me to make decisions about layout, and the flow of interaction. The two storyboards each flagged up different interface requirements, which I was then able to incorporate into the other one, using a concise number of elements to achieve the requirements of both scenarios.
Figure 7: Part of the storyboard for “from scratch” authoring of new documents.
5.1.3 Paper prototype

Many of the central interactions in the design that was now emerging were based on the 'drag-and-drop' paradigm, so a useful next step was to create a paper prototype, drawing each element on paper and cutting it out so that the pieces could be moved around. This allowed the rapid consideration of different layouts, as well as revealing unconsidered issues arising from the limits of laying out elements in 2D space.

Additionally, it provided a means for communicating my planned interface to my supervisors clearly, which was useful as they were able to help me identify potential improvements. The time investment for producing the paper prototype was minimal, and allowed me to fully consider every interface element that was to be implemented, providing a useful checklist for the requirements of the various frameworks, libraries and internal architecture to be selected.

5.1.4 Paper walkthrough

The final step in confirming the selected design was to perform a walkthrough of the two scenario archetypes with real data - the construction of a novel argument, and the markup of an actual essay using the tool. The aim of this stage was to confirm that the feature set selected encompassed all the required actions of argument construction and markup, and that no time would be wasted developing features that were redundant for real use cases.

This also flagged up elements of academic writing that had remained unconsidered up to this point, such as citations and URLs, providing a chance to consider their inclusion in the ontology of the graph design before beginning implementation and risking wasted development time.

5.2 Final design

5.2.1 Overview

Watch a demonstration video of the final design interface at http://youtu.be/W5uo2FBUL7c

The design of Write Reason is oriented around two panes: on the left, the document, and on the right, the graph. This is core to the goal of allowing structural visualisation and linear textual construction at the same time - by having both panes visible side by side, users can look at one pane without performing any actions, while interacting with the other pane. Further, this eliminates the need to hold these complex structures in the mind of the user - allowing cognitive power to be devoted to re-combining the ideas, incorporating new ideas, and formulating effective phrasing of the document. By making explicit the reasoning structure in a visual format, for larger and more complex
projects, the hope is that a richer and more accurate argument can be achieved, reducing the tendency to fall back on over-simplification due to the limits of working memory.

The grey divider between the two panes can be dragged from left to right, allowing the user to customise their workspace and give more screen real estate to one aspect while they focus on that. For example, if they choose to build a structural graph before starting to write, they can drag the divider all the way to the left to focus exclusively on the graph interface. This interaction pattern will be familiar to users of the \LaTeX{} editor Overleaf (overleaf.com).

5.2.2 Graph pane

A core design question was the ontology of the graph. I considered various formats - for example, having nodes of different kinds, such as nodes to internally represent a sub-argument in Standard Form \cite{2}. However, by introducing a more textured graph ontology, constraints to the use cases of the tool would be introduced, forcing users down a certain path. This is undesirable because styles of essay writing may vary radically between users and disciplines - to study the effect of graph visualisation \textit{in the general case} on argument construction, I decided to keep the tool as flexible as possible to differing methods. Thus, the chosen ontology is highly flexible.
Nodes  Nodes are added to the graph by double clicking on the background. By typing, the user can type the text which will label the node - a typically imagined case would be to write claims, propositions or pieces of evidence on the nodes, however due to the flexibility of the free text input they can be used to best suit the use case. Nodes can be freely moved around in the graph pane by dragging and dropping, allowing users to spatially organise their thoughts, thus building in further meaning and structure in an intuitive manner. To allow the graphs constructed to grow beyond the size of the graph pane, the viewport can be moved by dragging the background. Nodes can be deleted by right clicking on them and selecting “Delete” from the context menu.

Arrows  Edges between the nodes can be added by dragging from the area just outside a node, and dropping on another node. This action imitates that of drawing an arrow on paper, making the interaction familiar to users who have drawn mindmaps or flow charts in the past. Arrows are then anchored between the nodes, intelligently reflowing where possible to maintain visual distinction:

When an arrow is created, a modal dialog allows the user to quickly select from one of three arrow types: “Supports”, “Opposes” or “Expands”. This set of relations is intended to be sufficiently flexible as to be applicable to a wide variety of domains, supporting many different forms of reasoning. “Supports” and “Opposes” are self explanatory, while “Expands” is included as more of a wildcard relation, capturing cases of relations such as examples, related questions, referring back to earlier or later points, and anything else the user would determine as not falling under the other two categories. The arrow type is reflected in the colour of the arrow on the graph, giving a visual representation of the relation which can be observed at a glance. Arrow type can be edited by clicking on the
Figure 10: Two nodes connected with arrows in both directions.

arrow to allow change of plans as the user’s thought process evolves.

Figure 11: A modal dialog to select arrow type is displayed on arrow creation.

5.2.3 Document pane and interaction between panes

**Text editor**  The document pane’s primary role is as a typical text editor, providing a familiar interface for the construction of linear textual argument construction. Paragraphs can be inserted by pressing the return key, as usual. Each paragraph has a small grey box to its left, which can be dragged up or down to reorder paragraphs. This affords a much more natural method of revising the structure of a document than cutting and pasting, as a more visual method which does not require users to recall what content they have on their clipboard. This also reflects the method of reorganising the layout of nodes in the graph pane, providing a more cohesive mode of interaction across the two panes.

**Sections**  Nodes from the graph can be inserted into the document by dragging them from the left pane to the right pane at the desired location for the node to be inserted. I call the embodiment of a graph node in the document a
“section”. Sections are highlighted with a yellow background, and if the user hovers over a section, the corresponding node is highlighted in bright yellow. Similarly, if the user hovers over a node when it has a corresponding section, the section is highlighted in bright yellow. This creates a clear and continuous link between related aspects of the two panes, encouraging users to keep their plan and document in sync so that each is a faithful representation of the other, a primary goal of the tool. Graph nodes which are linked to sections are coloured red, while those not linked to sections are coloured blue, allowing users to easily see which parts of the graph have been added to the document.

To grant the user flexibility in the presentation of their document, sections can be given one of four styles.

- **Heading and body** Both the title and body are visible.

  - **The probability of existential risk is very low**
    
    A common argument against working on x-risk is that the chance is very low. However, it should be noted that if we did go extinct, the lost potential could be astronomically large (assuming you think the future utility of humans is net positive).

- **Heading only** The body text is hidden. This is useful for creating subtitles or the title for the whole document.

  - **We should be very concerned about existential risk**

- **Body only** The title text is hidden. This is useful for cases where the user doesn’t want to insert a title, but wants to link a paragraph with a graph node.
There has also been recent progress on better understanding the broader 'strategic' issues around AI. For instance, there has been research into how the government should respond to AI, covering arms races, the implications of sharing research openly, and the criteria on which AI policy is judged.

- **Inline** The section is rendered inline, like a `<span>` element in HTML, where other section styles are more like a `<div>` element. This allows users to relate specific words, phrases or sentences to nodes, allowing a more granular use of the graph to represent smaller reasoning steps. (Note that the image below demonstrates multiple different inline sections in a paragraph.)

By default, when a graph node is dragged onto the document, its style is “Headering and Body”.

Sections can also be created in the other direction - if the user types some text in a paragraph which is not currently a section, highlights it, then drags the highlighted text onto the graph pane, the highlighted text will be added to the graph as a node, and an inline section will be created wrapping the highlighted text, linked to the newly created graph node. This bidirectional ability to add nodes allows the updating of the plan in the graph as writing takes place, making it easy to keep the two panes in sync. Keeping the two panes in sync is made simpler by automated synchronisation in changes to headings: when the heading of a section is edited, the label on the node it is linked to is updated in kind, and vice versa.

### 5.2.4 Plain condition

A simple tool was developed for the other condition of the experiment - Write Reason is compared to a simple text editor with no graph visualisation, so that the effect of the graph can be observed. The text editor is basic, allowing typical interaction patterns, familiar to users of Notepad. Other word processing features were not added so that the only difference between the conditions is the addition of the graph visualisation, allowing its effect can be compared as directly as possible.

Whether a paragraph is a heading can be toggled by selecting some text and clicking the “Heading” button that pops up. Headings are displayed in bold.
and underlined, for consistency with Write Reason’s headings. The method of adding headings is necessarily different, because in Write Reason headings are added by adding a node from the graph to the document, and the plain condition has no graph.

Figure 14: Headings can be added by highlighting text and clicking “Heading”.

Figure 13: The plain text editor.
6 Implementation

6.1 Architectural decisions

Write Reason is a webapp built with React, using Slate.js (slatejs.org) for the document pane and SVG.js (svgjs.com) for the graph pane. A SharedState object manages the interaction between the two panes, which are implemented in the DocEditor and GraphPane. The DocEditor uses a stack of custom plugins for the Slate.js API, described below. The GraphPane manages a set of GraphNodes, and the connections between them, implemented using SVG.js and SVG.connectable.js (github.com/jillix/svg.connectable.js).

Webapp  Write Reason is built as a website, for flexibility of use - academic writing is performed in many contexts, across different operating systems, and in particular in contexts where users do not have root access and are unable to install new applications, such as on library or laboratory workstations. The website is exclusively client side, so could be easily extended to run offline as a Progressive Web App, or using Electron. Use of JavaScript has the advantage of granting access to the language’s excellent library support, which I hoped would support rapid development of this ambitious software project in the limited timeframe before running the study.
React

Facebook’s popular UI library React was chosen to underpin the structure of the webapp. I favoured React’s component-based approach because of the complexity of the application, and this helped me maintain well-separated components for more maintainable and readable JavaScript. Additionally, using React allowed the use of Slate, as described below. A further advantage of using React was providing an opportunity to develop my knowledge of the framework, as it is complex but powerful, and my prior experience was limited to one hackathon project.

Slate.js

I performed a comprehensive survey of the available frameworks for building rich text editors in JavaScript. There are a number of strong contenders, but what sets Slate apart from the alternatives is its customisability - I required deep integration with the graph pane, including novel components within the text area, such as sections. Slate’s idiomatic API was useful in managing the many problems that arise from customising HTML textareas - there are a very high number of ways that textareas allow users to interact, and features that users expect to be present in a text editing environment. As such, this presents a lot of scope for error, so judicious use of Slate was required to minimise these, preventing the text area from entering bad states and maintaining its tight connection with the graph pane.

I wrote a number of Slate.js custom plugins to augment the standard TextArea
functionality, adding the features of Write Reason and the plain text editor:

- **GraphPlugin** allows the user to drag graph nodes and drop them into the document
- **NodePlugin** manages the rendering and custom events for sections (the instantiation of graph nodes in the document)
- **ReorderPlugin** adds drag handles to the left of paragraph blocks, allowing them to be reorder by dragging and dropping
- **LoggingPlugin** detects cut, copy and paste events, and logs their type, time and content. This plugin is used in both Write Reason and the plain text editor
- **HoveringMenu** allows users to mark a paragraph as a heading, by displaying a button when some text is highlighted. This plugin is used only in the plain text editor

**SVG.js** The requirements for the graph pane required a flexible and high performance framework - nodes were to be moved around freely on a plane and labelled with text, which needed to be editable by clicking on it to select it. Arrows needed to be drawn between nodes. Finally, nodes needed the ability to be dragged off the edge of the graph, and dropped onto the document, as well as the opposite interaction: dragging snippets from the document onto the graph. SVG.js provided the flexibility needed, as a low level wrapper around JavaScript’s native SVG capabilities. SVG.js is not a React component, but I was able to incorporate it into the React webapp with few difficulties.

### 6.2 Difficult implementation details

**Library instability** A core difficulty for the stability of the webapp was Slate’s unfinished status - it is an open source project which is in a mature beta phase, but as I stacked a lot of customised features, I uncovered edge cases and issues which were difficult to work around. This was compounded by the fact that the library underwent a major update around two thirds of the way through my project, which updated many of the exposed functions and so it would have required a full rewrite to update to the new version. Instead, I stayed with the previous stable version, but this meant that the deep issues I uncovered were not fixed by the library’s maintainers, as they had moved onto maintaining the latest version. I was able to work around these difficulties by carefully avoiding features that were less stable, and adding explicit checks for breaking conditions.

This was a key learning point of the project for me - it might have been less time consuming to do more of the low level work myself and just use the native...
HTML textarea element instead of Slate’s abstraction over it, as I would have had a stronger understanding of the structure, and would not have been affected by external issues. On the other hand, it was a good learning experience to use a well designed JavaScript API (despite its instability), and this supported good structure in my own code, which contributed to finishing a fully featured prototype in time for the study.

### Implementing drag and drop support

One specific interesting challenge was implementing drag and drop support across the two panes. HTML5 has an excellent Drag and Drop API for transferring data between parts of a webapp, and between a webapp and other applications. However, this API does not support SVG DOM elements, which was the basis of my graph pane. I therefore implemented drag and drop support myself, which was an interesting challenge, using DOM mouse events and custom events for tracking which node was being dropped, and where to insert the dropped text in the document pane.

This additional effort was worthwhile, as it provides a unified interaction pattern across the two panes, where users can drag to reorder document paragraphs, drag to move graph nodes, and drag nodes and highlighted text from one side to the other to insert them. When showing study participants the instructional video for Write Reason, many of them commented that they liked the ability to drag nodes from the graph pane to the document pane when this feature was demonstrated.

### Logging

Detailed automated log generation was implemented, whereby almost all actions performed by users of both the Write Reason and the plain condition tool are tracked as log events. All events have a type and timestamp, and specific event types have additional properties - for example, the “paste” event also holds the pasted content in plaintext and HTML. Logs were saved in Session Storage to preserve the data in case of early termination of the experiment (such as by accidentally refreshing the page), and on completion of a condition the logs were compiled into a JSON file and downloaded for later analysis.

### 6.3 Testing

Testing was performed continuously, by repeated manual testing of the interface. While solutions exist for unit testing of the user interface of web applications, the complexity of the interactions in this project (particularly text editor interaction and drag and drop actions) prohibits the utility of these solutions - the development cost of setting up automated solutions outweighed the benefits in this case, given the tight timeframe. Testing was performed by exercising each of the interface components in the browser, presenting cases of each different input, in isolation and in combination with other interactions. In particular,
the text editor component required heavy testing due to the number of edge
cases around node ordering, document content, cursor placement, and modes
of interaction (such as pasting, cutting, typing, and so on). Another source of
potential problems, which I therefore tested heavily, was interactions between
the document and graph panes which involve the modification or deletion of a
node in one pane, then performing operations on its linked node in the other
pane. At the end, I ran a final focused phase of trying to find breaking edge
case interactions before running the study.

Facebook’s Create React App library [create-react-app.dev] was used to manage
project initialisation, and supports live updating of the local instance of the
webapp as changes are made to files. This allowed extremely fast iterations of
development and testing. Judicious use of typical web development tools such
as Chrome’s Developer Tools were invaluable in the debugging process.

7 Study method

The core of the project was the design, execution and analysis of a user study.
The study aimed to shed light on the question:

*How is the quality of academic writing affected by visualising its reasoning structure?*

This is part of a broader research area, which examines how tool use can aid
construction and comprehension of arguments. To determine the effect of a
reasoning structure graph, two conditions were required:

- One in which participants use a tool with a reasoning structure graph,
  Write Reason
- One in which participants use a tool without a reasoning structure graph

These conditions will be referred to as “graph” and “plain”, respectively. The
central hypothesis is that graph visualisation improves both the quality of essay
construction and essay comprehension, and Write Reason was designed with
both use cases in mind - supporting both the “from-scratch” authoring of new
essays, and the mark-up of existing documents to understand and critique their
underlying reasoning structure. However, these two divergent questions were
too much to cover in one study, so this study was aimed at examining the
“from-scratch” authoring case.

Various study designs were considered, but I settled on one which aimed to ex-
ercise the two tools in this “from-scratch” case as straightforwardly as possible,
by asking participants to use each of them in turn to construct an essay, in order
to observe the effect with as few confounds as possible. The essays constructed
were then scored by other participants, “markers”, on clarity, persuasiveness, structure, objection responsiveness and an overall evaluation, so that the effect of the two tools on these topics could be measured. These measures are discussed further in Subsection 7.6.

7.1 Topic selection

A key consideration for the design of the study was the topics on which participants were asked to write their essays. High variation in essay topics could represent confounds for the measurement of essay quality - for example, certain topics may be less complex, thus not showing differences in tool effect which only present when discussing more complex topics. Alternatively, topic selection may affect scores assigned by markers, depending on the marker’s interest and knowledge level about the topic. To minimise the effect of topic selection, thereby making more clear the effect of the tools used, I asked participants to write on specific topics.

The two topics selected were:

- Genetic biohacking, which is genetic experimentation taking place outside of research and industrial settings
- Shared spaces in urban design, a design approach which involves the elimination of traffic controls such as signs, kerbs and pedestrian crossings

The topics were selected based on a number of criteria. Firstly, if some participants knew a great deal about the topic while others knew very little, then this may obscure the effect of tools, so to minimise the chance of this I selected topics which were relatively obscure. These topics adhered to our criteria because the questions I gave participants are not settled and there are proponents for both sides of the arguments, which makes it a more complex task, better exercising the tools. This also gives the experiment better external validity, because academic writing often consists in contributing to a debate or providing additional evidence for a topic that is not settled, and frequently involves the examination of multiple plausible solutions to or perspectives on a problem. Another consideration in topic selection was ethical in nature - it was important that the topics were unlikely to relate to particularly controversial and upsetting content, in particular avoiding any areas which could remind participants of personal trauma. I selected biohacking and shared spaces as they met all of these conditions, and I hoped that they would be interesting areas for participants to spend some time writing about.

For each topic, participants were asked to respond to a question:

- For the biohacking topic, the question was “Should greater regulatory control should be exerted over biohacking?”
• For the shared space topic, the question was “Should shared spaces in urban planning should be promoted?”

7.2 Fact sheets

To further minimise the variance in relevant knowledge amongst participants, I provided all participants with the same information on the topic. The information was given in the form of a “fact sheet” - a document containing a set of facts relating to the question.

I considered giving participants an introductory article to read on the topic, but this would likely (explicitly or implicitly) contain an argument for some position on the question at hand, thus reducing the complexity of the task, as participants could simply reproduce the existing argument. This would run contrary to the aim of the experiment, which relates to the construction of arguments in essays. To counter this, I considered giving participants multiple articles to read, each presenting different viewpoints on the question, to lead them to construct new arguments building on the arguments in each article. This approach has good external validity - real academic writing often involves reading multiple arguments and synthesising a view from them. However, a practical limitation was the constraint on the amount of time that could be reasonably asked of participants, and reading multiple articles on two different topics would have taken too much of the study time away from the task of essay construction. My solution was to construct fact sheets with excerpts from multiple sources. The full fact sheets can be read in Subsection B.

The motivation behind the fact sheets was to focus the study specifically on the essay construction phase of academic writing (not other phases such as finding relevant literature or reading and note-taking). By asking participants to read a collection of quotes from articles on the topic, I hoped to imitate the effect of having read larger articles on the topic and remembered or written down relevant quotes - while eliminating the actual execution of this part of the process from the study. This would allow participants to spend as much of the time as possible on the essay construction itself.

Two fact sheets were compiled, one for each topic. Each fact sheet contained two quotes neutrally defining the topic, one quote neutrally giving an example of the topic, five quotes arguing we should answer “yes” to the question, and five quotes arguing we should answer “no”.

The ordering of the quotes was carefully managed to prevent a clear argument being presented by the fact sheet itself, which might occur, for example, if quotes were ordered such that one quote raised an objection and the following quote responded to it. Additionally, the fact sheets were intended to be informational and balanced, presenting both the pro and con cases. Therefore, the ordering of the sheets was semi-randomised, with quotes alternating between pro and con, and quotes on similar parts of the issue were not presented consecutively. The
definitions and examples were scattered throughout this ordering of quotes. The ordering thus avoids the fact sheets presenting the pro quotes first followed by the con quotes, or vice versa, to avoid nudging participants towards constructing an essay arguing for a particular side of the debate. This gave participants the agency to freely select a position on the topic and argue for it with a range of evidence, as well as responding to opposing views presented on the fact sheet - all aiming to reflect that these properties which are often present in academic writing.

At the top of the fact sheet is the question for participants to respond to in their submitted essay, and an instructional line: “Please write a response to the question below. Write your response in prose, in the style of a short essay.” This instruction serves as a reminder of the instructions I gave to participants verbally, aiming to minimise the chance of participants misinterpreting the task leading to them writing in a non-academic style, which would be less relevant to the research question as it focuses on academic writing.

7.3 Participants and conditions

24 participants were recruited by email, social media, flyers in university buildings, and word of mouth. The participants consisted in 12 undergraduate, and 12 postgraduate students (11 male, 11 female, 1 non-binary, 1 prefer not to say) at the University of St Andrews. Students were the chosen population as they regularly perform academic writing. I was interested in understanding the effects of degree level upon the overall study question, so the participants were equally divided between these two conditions.

I opted for a within-subjects design, in which each participant completed both the graph and plain conditions. This was on the basis that I expected high variation in individual ability in essay writing, and so by having each participant perform both tasks the effect of underlying ability could be minimised by analysing the difference in scores and other properties of essays constructed with the two tools. Additionally, due to the difficulty of sourcing large numbers of participants, this allowed a greater number of data points for each condition, generating more confidence in the results of the study.

Due to the nature of the study, there are a number of different possibilities for each participant. They can be undergraduate or postgraduate, they can do the shared space question first, or the biohacking question first, and they can do the graph or plain task first. An experimental condition, therefore, consists in three components: degree, topic and tool. There are 8 experimental conditions. To be able to understand and eliminate each of these effects, and to get balanced results, participants were randomly allocated to conditions. There were 24 participants, so three participants were assigned to each condition, as shown in Figure 17.
7.4 Experimental procedure

For internal validity, I kept the process of each of the 24 experiment sessions as similar as possible. Write Reason was designed to manage some of this, by tracking timings and transitions between sections of the experiment. All studies were conducted in the same room in the School of Computer Science, with a specially prepared computer. I used a script to make sure the instructions I gave to each participant were the same, for internal validity. The experiments followed the following structure:

- **Introduction**
  I described the study to participants and showed them the participant information sheet. I answered any questions, and they signed a consent form once they were happy to proceed.

- **Familiarisation** (3 minutes)
I showed participants a short instructional video demonstrating how to use the features of the tool. I then gave them three minutes to familiarise themselves with the tool, in a sandbox mode where their changes were subsequently discarded. I answered any questions on tool usage they had in this section, and during the essay construction phase.

- **Fact sheet reading** (5 minutes)

  I gave participants the fact sheet for the topic, pointed out the question that they would subsequently be answering, and asked them to take 5 minutes to read the fact sheet. I reassured them that we expected they would likely know little about the topic, and might find the fact sheet confusing - and that this was part of the design of the experiment. Participants were not allowed to use the tool or otherwise make any notes: these 5 minutes were just for the purpose of reading the fact sheet.

- **Essay construction** (15 minutes)

  I then asked participants to spend 15 minutes constructing a short written essay answering the question on the fact sheet. I asked them to write in prose, in an academic style. I informed them that I would let them know when there were 5 and 1 minutes remaining in the task. I asked them to exclusively use the fact sheet, their own existing knowledge, and the software tool, and to not enter any personally identifying information. They retained access to the fact sheet to refer back to during the essay construction phase.

- **Repeat for other condition**

  Each of the above phases except for Introduction were repeated for the second condition. The second condition was either the graph or the plain task, as determined by the balancing of participants between different conditions described in Subsection 7.3. In between conditions I gave participants the opportunity to take a five minute break.

- **Questionnaire**

  Finally, I gave participants a short questionnaire to complete, collecting basic demographic information, impressions of the two tools they had used, and prior experience with essay writing software. Participants were given a £5 Amazon voucher to thank them for their time.

During the essay construction phase for the plain condition, I placed a blank A4 piece of paper and a pen on the table beside the participants, which they were able to use for making notes, planning or any other purpose. The motivation for this was to not artificially constrain participants to a text editor alone, because in almost any real scenario the use of paper would be a possibility. Additionally, I was interested to observe whether participants would use the paper to draw graphs similar to those they constructed in Write Reason, or for some other use. I did not instruct them to use the paper or alert them to the possibility of using
paper because I did not want to bias them towards using paper to plan if they
would not have done so otherwise.

7.5 Apparatus

The study was conducted on a desktop computer, with a conventional free-
standing monitor, mouse and keyboard. Write Reason and the plain text editor
were presented in Google Chrome, and the transition between the two conditions
and their respective sandbox modes was controlled by the application. Users’
screens were recorded using the Screen Recorder Chrome extension. The fact
sheets were printed on double-sided A4 paper, laminated.

As much of the work as possible of running the experiment was offloaded onto
the software tool, with the aim of making the 24 sessions with participants as
similar as possible, to avoid confounds.

Before beginning the experiment, the conditions and an experiment ID are en-
tered so that they can be stored in the logs.

![Figure 18: Experimental controls.](image)

A sandbox mode is provided for the Write Reason and plain conditions. The
sandbox mode does not generate logs or save the essay, and displays a 3 minute
timer, for the purposes of familiarisation (see section 7). Upon exiting sandbox
mode, the tool is stopped, and the full 15 minute timer is started.

![Figure 19: The timer, alongside a button to move to the next phase of the
experiment.](image)

The participant’s screen was recorded throughout, and both the Write Reason
tool and the plain text editor logged almost all actions taken by the user.
7.6 Marking

To provide a basis for comparison of the kind and quality of essays produced, the essays were marked by other participants, markers, on several marking criteria: clarity, persuasiveness, structure, objection responsiveness and overall. I recruited three PhD students in philosophy to perform the marking, because philosophers are trained experts in essay construction and argumentation, and because PhD students who perform tutoring, as all three of the markers in this study do, will frequently perform marking for undergraduate Philosophy coursework.

These factors increased confidence that the scores assigned reflect the underlying properties (clarity, persuasiveness, etc) which I was interested in studying. Each essay submitted was marked by all three markers, to allow reduction in the effect of outliers by averaging marks.

The marking criteria were selected to cover different aspects of essays, all relevant to their quality.

<table>
<thead>
<tr>
<th>Marking descriptor</th>
<th>1-3 (Poor)</th>
<th>4-6 (Middling)</th>
<th>7-10 (Strong)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clarity</td>
<td>Poor clarity – claims are hard to understand</td>
<td>Middling clarity – claims are reasonably well described</td>
<td>Strong clarity – claims are articulate and clear</td>
</tr>
<tr>
<td>Persuasiveness</td>
<td>Poor persuasiveness – argument carries negligible force or is invalid</td>
<td>Middling persuasiveness – argument is fairly compelling though not watertight</td>
<td>Strong persuasiveness – argument is highly compelling</td>
</tr>
<tr>
<td>Structure</td>
<td>Poor structure – poorly organised argument that is difficult to follow</td>
<td>Middling structure – somewhat logically organised argument</td>
<td>Strong structure – highly effective organisation that presents the argument powerfully</td>
</tr>
<tr>
<td>Objection responsiveness</td>
<td>Poor objection responsiveness – no consideration of any opposing view</td>
<td>Middling objection responsiveness – some consideration of opposing view, with weak response</td>
<td>Strong objection responsiveness – good description and response to opposing view</td>
</tr>
<tr>
<td>Overall assessment of argument</td>
<td>Poor overall</td>
<td>Middling overall</td>
<td>Strong overall</td>
</tr>
</tbody>
</table>

Figure 20: The full description of score bands for each of the marking criteria, given to the markers.

The four specific criteria were selected for good coverage of different aspects of the essays: clarity focuses on individual claims, structure focuses on the way claims are combined, persuasiveness captures the rhetorical force of the presentation, and objection responsiveness captures the description and response to opposing views. Finally, the overall score provides a summary of the expert
judgement of the markers, allowing a more holistic scoring of the essay.

The ordering of the essays given to the markers was semi-randomised. Each marking document contained the 48 essays, alternating between essays constructed in the plain condition and the graph condition. For two markers, the first essay in the document was constructed in the plain condition, and for the third, the first was constructed in the graph condition. The order of the essays within the alternating pattern was shuffled for each marker, so the three markers marked the documents in different orders, to mitigate effects upon the scores from ordering.

Markers were instructed to focus on the argumentative quality of the essays, and keep other considerations out of judgement, such as grammar, spelling and length. The essays given to markers were anonymised, and the only text accompanying them was the question which the essay was a response to - the markers did not know which essays were constructed using one tool or the other.

8 Analysis

Throughout this analysis section, where a marking criteria (clarity, persuasiveness, structure, objection responsiveness or overall) is referred to, note that for each submitted essay, the marks from the three markers were averaged, to gain an aggregate of the three expert views.

8.1 Difference in scores between tools

My initial hypotheses about the difference in scores amongst tools are shown below:

**H1** Scores will be higher for essays in the graph condition than the plain condition

**H1.1** Clarity scores will be higher for graph than plain.

**H1.2** Persuasiveness scores will be higher for graph than plain.

**H1.3** Structure scores will be higher for graph than plain.

**H1.4** Objection responsiveness scores will be higher for graph than plain.

**H1.5** Overall scores will be higher for graph than plain.
H2 Word count will be higher for plain than graph.

The mean scores attained under each of the two conditions (plain and graph) are in displayed Table 1.

<table>
<thead>
<tr>
<th>Marking criteria</th>
<th>Plain</th>
<th>Graph</th>
<th>Dependent t-test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clarity</td>
<td>6.097</td>
<td>5.792</td>
<td>t(23)=0.979, p=0.338</td>
</tr>
<tr>
<td>Persuasiveness</td>
<td>5.306</td>
<td>4.403</td>
<td>t(23)=2.338, p=0.028</td>
</tr>
<tr>
<td>Structure</td>
<td>5.764</td>
<td>5.458</td>
<td>t(23)=0.927, p=0.364</td>
</tr>
<tr>
<td>Objection responsiveness</td>
<td>4.292</td>
<td>3.778</td>
<td>t(23)=1.231, p=0.231</td>
</tr>
<tr>
<td>Overall</td>
<td>5.264</td>
<td>4.667</td>
<td>t(23)=1.612, p=0.121</td>
</tr>
</tbody>
</table>

Table 1: Average scores for each condition, and the results of a dependent t-test.

Using a dependent t-test (because this is a within-subjects design, see Subsection 8.3 for justification), this shows that performance was significantly better for the plain condition than the graph condition for persuasiveness, as shown in Table 1. For other scores, no significant effect was found - for all t-tests, p > 0.05. Therefore my hypotheses H1.1, H1.2, H1.3 H1.4 and H1.5 are not supported - in fact, the opposite of H1.2 was shown by the data for persuasiveness.

My hypothesis H2 was confirmed. The average number of words in essays constructed in the plain condition was 274.9, and in the graph condition 221.6, a difference of 53.3. This effect was significant (t(23)=3.498, p=0.002). This is reflected in Figure 22, where you can see that submissions in the plain condition tended to be longer than those in the graph condition.
8.2 Score difference between participant groups

My initial hypothesis about the difference in scores amongst participant groups was:

**H3** Postgraduates will attain higher scores on average than undergraduates for all marking criteria

The participants recruited were balanced between undergraduate and postgraduates, so that the effect of level of study upon scores attained under the two conditions could be examined. Surprisingly, hypothesis H3 was not supported - the data shows that postgraduates achieved significantly lower scores for persuasiveness, $t(23)=-2.962$, $p=0.005$, and overall, $t(23)=-2.864$, $p=0.006$. These hypotheses were tested using independent t-tests, because the undergraduate and postgraduate participants are two distinct sets.

Post hoc analysis was performed to determine whether self-reported frequency of writing essays, in the questionnaire, had any effect on scores. Options on the questionnaire were daily, weekly, monthly, rarely or never. For comparison, these were bucketed into commonly (daily or weekly, $n=14$) and uncommonly (monthly, rarely or never, $n=10$). No significant effect upon score attained was found between these two groups, contrary to expectations that more frequent essay writers would gain higher scores.
8.3 Testing assumptions of t-tests

Figure 23: Distribution of scores achieved.

In the previous two sections, I used a dependent t-test for differences in scores across conditions, and an independent t-test for differences in scores across participant groups. In this section, I show that the assumptions of these two tests are met by the data, and so they are appropriate hypothesis tests to use. For the independent t-test an assumption is that the dependent variable is approximately normally distributed, which is shown to be the case by visual inspection of Figure 23.

For the dependent t-test, an assumption is that the distribution of differences in the dependent variable is approximately normally distributed. Figure 24 demonstrates that this assumption holds. Additionally, I ran a Shapiro-Wilk test for normality for the differences in scores attained. The null hypothesis of the test is that the population is distributed normally. For all marking criteria, the p-values were much greater than 0.05, meaning that we cannot reject the hypothesis that the population is distributed normally.

8.4 Tool preference

In the questionnaire, participants were asked “Given the choice of using the two tools you used today to do some academic writing, which would you choose?”. 16 preferred Write Reason, 8 preferred the plain text editor.
Post hoc analysis initially seems to suggest that preference was influenced by the order in which the tools were presented. Figure 26 compares participant preferences for the graph and plain tools grouped by whether they used the graph tool first or second.

However, it is not clear that ordering causes this effect. Participants studying Computer Science, who made up 9 out of the 24 participants, were much more likely to prefer the graph condition than other participants, and, by chance Computer Science students used the graph tool first only 3/9 times.

Thus the data is not conclusive, but might suggest that Computer Science students are more likely than other students to prefer Write Reason. It also might suggest that the tool used second has some influence over which tool is preferred, perhaps due to psychological recency effects. Further study would be required to draw stronger conclusions about these effects and their mechanisms.
8.5 Graphs constructed

Having analysed comparisons between Write Reason and the plain condition, I now focus on examining how participants used Write Reason in particular. In the experimental design, I made sure not to enforce or encourage any given usage pattern of the tool, and participants were informed that the graphs generated would not be given to the markers, only the essays they write would be. Thus this provides an opportunity to examine how participants react given the option to use a graph-based interface, and what they use it for. This is a novel contribution to the literature, as prior work has typically enforced usage of argument graphs.

The kinds of graphs generated by users were analysed qualitatively to look for common strategies and uses. The categories of graphs found in the data are displayed in Table 2.

6 of the 24 participants elected not to use the graph at all. All other participants used the graph to plan their essay, adding nodes before beginning to write.
Table 2: Descriptions of graph categories observed.

<table>
<thead>
<tr>
<th>Categorisation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>no-use</td>
<td>Graph is not used at all.</td>
</tr>
<tr>
<td>pro-con-centralised</td>
<td>Central node is the question or central claim. Pros and cons are connected</td>
</tr>
<tr>
<td></td>
<td>to it with supports/opposes arrows. Each node is connected to every other</td>
</tr>
<tr>
<td></td>
<td>node at least indirectly.</td>
</tr>
<tr>
<td>no-arrows-notes</td>
<td>Nodes are (almost) totally disconnected. Nodes are used to briefly summarise</td>
</tr>
<tr>
<td></td>
<td>points to cover in the essay.</td>
</tr>
<tr>
<td>paragraphs</td>
<td>Full paragraphs are written in nodes then added to the graph. Nodes are</td>
</tr>
<tr>
<td></td>
<td>largely disconnected.</td>
</tr>
<tr>
<td>pro-con-separated</td>
<td>Primarily, two separate subgraphs, one for pros and one for cons. Sometimes,</td>
</tr>
<tr>
<td></td>
<td>some additional disconnected small subgraphs are present.</td>
</tr>
<tr>
<td>evidence-linking</td>
<td>Nodes describe different bodies of evidence, arrows are used to indicate</td>
</tr>
<tr>
<td></td>
<td>which support/oppose each other.</td>
</tr>
<tr>
<td>misc</td>
<td>Miscellaneous approaches.</td>
</tr>
</tbody>
</table>

Figure 27: Count of graphs observed by category.
A variety of usage patterns of the graph were observed. The most common strategy was pro-con-centralised, an argument mapping approach reminiscent of mindmapping. In this strategy, nodes are arranged around a central node which is labelled with the central claim or topic. Pros are connected with green “Supports” arrows and cons are connected with red “Opposes” arrows. Most of the graphs observed were shallow, with most nodes connected to the central node having no children.

Figure 28: An example of a pro-con-centralised graph.
Another usage pattern was pro-con-separated, in which most of the graph is dedicated to two disconnected subgraphs: one describing the arguments for the claim, and the other describing arguments against the claim. In Figure 29, three pros and one con are presented. Interestingly, the participant used a node to describe a piece of evidence against shared spaces (“the Holmes report shows large margin of dissatisfaction”), but then uses another node as an attack - “however no indication of sample size or make up, not the best evidence” - and links this node to the positive claim - “should use shared spaces”. This shows a nuanced usage of the tool for essays with objections and replies, demonstrating the potential of a graph for deeper conceptual understanding of a topic which goes beyond a simple list of pros and cons.

Figure 29: An example of a pro-con-separated graph.

A different approach to the use of the graph was observed in the case of evidence-linking. While only one user employed this strategy, I highlight it because it represents a different approach to the graph’s usage, presenting interesting further questions. In this strategy, shown in Figure 30, the participant has used the nodes to represent each of the quotes given to them on their fact sheet, using the sources of the quotes and a short summary label to identify them. The participant then added connections between these different quotes reflecting whether they support, oppose or expand on each other.

Due to the complexity of this strategy, which involves reviewing all of the provided material multiple times to find different connections, it seems that the participant had not finished building the graph, but one possible next step would have been to link the unused “QUESTION” node to evidence nodes to
build chains of arguments. Of these, some could then be selected for insertion into the essay, thereby allowing the linearisation of this network of evidential relations. This is an interesting approach and I expect that given more time and a greater evidence base, this could be a useful approach to making sense of the relations between pieces of evidence.

![Evidence-linking graph](image)

Figure 30: A evidence-linking graph.

Post hoc analysis of the overall scores shows that users who used the pro-concentrised approach, or did not use the graph at all, tended to score the highest overall, as Figure 31 shows. Other strategies generally scored lower, although for most of them there are only one or two data points so no particular conclusions can be drawn.

There is potential for further study of the tool in which participants are given more time for a task, or multiple tasks, across which they can develop a more optimal personal approach to the tool. Data about usage of more experienced
users with the time to consider their approach more carefully may show a positive effect upon the quality of essays constructed as a result.

8.6 Analysis of scores

Due to the subjective nature of essay evaluation, it is important to examine the level of agreement amongst the markers in scores assigned, to determine the level of confidence warranted in the scores assigned. The 3 markers were selected as experts in argument evaluation, as philosophy PhD students with experience marking undergraduate essays, which supports the adoption of a prior that their evaluation of arguments is likely to be accurate. Figure 32 shows the Pearson correlation between each of the marking criteria for each of the markers. It also shows the Pearson correlation of each of these measures with the number of words in the essay (arg.words) and the number of nodes added to the graph (total_nodes):

Figure 32 shows a significant positive correlation between almost all marking criteria for each of the 3 markers. Each marker’s assigned scores tends to corre-
Figure 32: Pearson correlation between different markers’ assigned scores. All reported values are significant ($p < 0.05$), non-significant results are omitted. Note that “objection resp” stands for “objection responsiveness”.

late most heavily with the other scores assigned by that marker. In particular, each marker’s “overall” scoring correlates very heavily with their “persuasiveness” scoring - correlation of 0.96, 0.9 and 0.95 respectively for markers 1, 2, and 3.

The “objection responsiveness” scores were found to not have weak or no significant correlation with the other scores assigned. Due to the short length of most responses (average word count was 248), there was little scope for the presentation of opposing views and response to them. Thus, perceived variation in objection responsiveness scores may have been due to implicit suggestions of opposing views and response to them, being more subjective and thus having higher variance between the markers. This suggests that strong conclusions about essay quality should not be drawn on the basis of the objection responsiveness scores, due to a higher level of disagreement amongst the experts. For this reason, in the analysis above, objection responsiveness scores have not been
used to draw any substantive conclusions.

On the other hand, because the other marking criteria (clarity, structure, persuasiveness and overall) exhibit a reasonable positive degree of correlation between markers, this suggests these scores provide a good basis for drawing conclusions about the quality of the essays. Figure 32 also demonstrates that most measures had a moderate positive correlation with essay length (arg_words), and a moderate negative correlation with the number of nodes added to the graph (total_nodes). This is expected given the results across the two conditions described in Subsection 8.1

8.7 Use of paper

In the study design section, Subsection 7.4 I described the fact that I gave participants pen and paper to make notes during the plain condition if they wanted to. Only one participant made use of the paper, and they only wrote three words at the beginning of the task, then used the tool exclusively.

8.8 Analysis methodology

Analysis was performed in Python using pandas and scipy, with Matplotlib and Seaborn for visualisation, and Jupyter Notebook for interactive data exploration. The results can be reproduced using the iPython notebook file located at analysis/analyse.ipynb.

9 Discussion and limitations

The project’s central research question was:

How does graph-based visualisation of reasoning structure affect argument construction in academic writing?

While this broad research question cannot be answered fully in one study, this work presents a partial answer. The design of Write Reason and this study begin to answer this question, by demonstrating how graph-based visualisation of the kinds constructable in Write Reason affects the scores attained in this study’s short essay writing task.

Design of Write Reason A novel contribution of this work is the design of Write Reason, a tool combining a text editor and an interface for argument mapping. The tool is flexible in its usage, allowing the user to select their preferred strategy for the task at hand, because its graph ontology is open
ended, and imposes little structure. This flexibility situates it as well suited to answering the research question, which concerns academic writing as a whole: a domain containing a large variety of styles and methods.

**Study design** The user study addressed the research question by having 24 participants perform an argument construction task with and without the graph-based visualisation of Write Reason. The results were unexpected. Persuasiveness was found to be significantly higher in essays constructed using the plain text editor than Write Reason. Other scores (clarity, structure, objection responsiveness and overall score) were not found to be significantly affected by the tool used.

**Scores attained** The study’s results demonstrate that this topic is a complex one. Many factors influence the quality of academic writing, and these are highly contextual, introducing a great deal of noise. Persuasiveness stands out as the criteria found to be negatively affected by the use of Write Reason. An explanation for this is that participants write more persuasive essays if they spend more time crafting the text of their essay, as opposed to gaining a structural understanding using the graph.

**Graphs constructed** A key contribution of this work was the opportunity to observe graphs constructed by participants when allowed to choose a strategy that suited them best. As noted in Subsection 8.5, 6 out of the 24 participants elected not to use the graph. Based on questionnaire feedback, it seems that one cause of this was confidence in a non-graph based typical planning and writing style - it had served them well in the past, so they did not want to deviate from that technique because a graph might be less effective for them. However, a further question is whether this cautious approach is warranted. Users who elected not to use the graph in this short study attained amongst the highest scores. This may reflect a confidence in their essay writing abilities, which leads them to stick to their own typical approach, and is warranted as they are good at writing essays.

As discussed earlier, post hoc analysis of the overall scores shows that users who used the pro-con-centralised approach (or did not use the graph at all), tended to score the highest overall. Mindmapping is a commonly used technique, and the pro-con-centralised strategy is strongly reminiscent of the mindmapping technique, with the addition of “supports” and “opposes” arrows to denote pros and cons. Therefore, participants using the pro-con-centralised strategy would have been able to apply it with minimal thought going into the development of the strategy itself. On the other hand, more novel uses of the graph may have been more effortful to execute, because they require many decisions about how the various elements (nodes, arrows of different kinds, spatial arrangement, and so on) are to be used, therefore leaving less time for the construction of the essay. Another explanation is that users of more unusual strategies may have
been unsure how best to use the tool, and therefore did not succeed in using it as effectively.

9.1 Limitations and future work

Short task duration A limitation of this work is the 15 minute duration of the essay writing tasks. This limits the external validity of the results, as typically academic writing is an extended process taking place over a much longer period of time. However, the results demonstrate the complexity of this research question, and can generate numerous hypotheses leading to opportunities for further study.

Effect of conditions upon scores Further study is required to fully understand the cause of the differences in scores attained between conditions. My leading explanation for why these unexpected results were found is due to the task’s short duration. In the plain condition, they are using an interface that is highly familiar to them, allowing them to immediately get started on the challenge of writing this essay. However, in the graph condition, they are getting to grips with a new user interface and, for many participants, a wholly new way of writing essays. As a result, a possible explanation is that participants spent too much time becoming accustomed with the new interface and building the graph, thus leaving little time to communicate their ideas in the essay. This is reflected in the lower number of words written in the graph condition. This explanation is also supported by participant responses in the questionnaire - for example, “the graph interface would be useful for an actual academic essay but took up lots of time in this fast setting”, “making the graph was time consuming”, “I spent more time drafting my ideas on the graph than I should have”. This hypothesis could be tested by a further study evaluating Write Reason that would run over a longer timeframe.

Effect of degree upon scores Postgraduate students were found to perform significantly worse than undergraduate students for persuasiveness and overall evaluation. Additionally, no significant effect was found of self-reported essay writing frequency upon score. Further study is required to determine the causes of these unexpected results. This raises interesting questions around the specificity of postgraduate expertise - for instance, it may be that the additional academic training postgraduate students receive mostly affects their writing quality only in their chosen subject area.

Graphs constructed In repeated use of Write Reason, users could develop different strategies for graph usage. Novel approaches, such as the evidence-linking approach discussed in Subsection 8.5 may have an effect upon essays generated, compared to a more typical approach such as pro-con-centralised.
Further study could explore these questions, in particular examining whether the most effective approach is idiosyncratic to individuals, or whether certain patterns are most effective across users. Another further question is how these approaches interact with collaborative projects - it is unclear whether the approach most effective for individuals would also be the most effective approach for a group setting. Additionally, differences in the style of output desired, as well as the domain, may affect which strategy is most effective.

10 Evaluation

The project’s performance against each of the revised requirements is laid out below:

(P1) Design and build an application which combines text editing facilities with tools to plan, communicate and recall a graph of the reasoning structure

\textit{This requirement was completed in the production of Write Reason.}

(P2) The application supports “from scratch” authoring of projects by planning the structure of the piece and subsequently filling the prose, while maintaining and updating the structural map

\textit{Write Reason was designed with this use case in mind, and was shown to succeed by its use for this scenario in the user study.}

(P3) Conduct a user study to determine the effect of the application on the individual ideation and writing process

\textit{The user study was completed as described in section 7.}

(S1) The application supports mark-up of existing pieces of academic writing, to extract structure and aid understanding and critical analysis

\textit{While this was not tested in the user study, this requirement was completed - the text from an existing document can be copied into Write Reason, then the graph can be used to analyse it.}

(T1) The application supports real-time collaboration between different users in the same project

\textit{This tertiary requirement was not completed, in favour of focusing more project time on the user study design, execution and analysis.}

(T2) The application supports templating for graph node types

\textit{This tertiary requirement was not pursued, as the final design aimed at flexibility of use rather than prescribing usage via the node types.}

(T3) The application supports templating for graph structures

\textit{This tertiary requirement was not completed to focus development time on features relevant for the study.}
(T4) Release the tool to the world to see if it’s picked up by users

The Write Reason code submitted in this dissertation is set up specifically for running the experiment. However, I plan to perform the required revisions and publicly release a stable, complete version, with some additional features not required for the experiment, in the near future.

The project was a success, having met all of the primary and secondary requirements. The user study generated a great deal of interesting data, only some of which was used in the analysis above - there is scope for insights from further analysis, particularly in conjunction with follow up studies to clarify causal mechanisms. While the user study did not find a positive effect of Write Reason on essay quality as judged by markers, it generated a number of other interesting results and clarified what kind of future study which would be a stronger test of the efficacy of the tool.

The design and implementation of Write Reason is a novel contribution to the literature, as no prior work has involved the creation of software combining a linear text editor for essay writing with a graph view for argument mapping, in a single piece of software. The user study is also a novel contribution as it tests the effect of the new software, and in particular because it tests how participants choose to use an argument mapping interface to represent their ideas when no constraints are imposed upon them, where previous work tests the effect of instructing participants to use or evaluate particular methods. Additionally, as far as I know, Write Reason presents a featureset not available in any commercially available software package, as described in Subsection 2.3.

11 Conclusions

This project presents the design, implementation and evaluation of Write Reason, a novel software tool for academic writing, combining a text editor with a graph interface for argument mapping.

The user study showed that essays written using a plain text editor are significantly more persuasive than those written using Write Reason, under the time-constrained conditions tested. No significant effect was observed on clarity, structure, objection responsiveness and overall scoring. Word count was significantly higher when using the plain text editor than when using Write Reason. This supports the hypothesis that under a short 15 minute time constraint, participants new to Write Reason spent a large portion of their time mapping out the reasoning structure, leaving little time to write an essay, and therefore gained lower scores than those using a familiar text editor tool. This area is ripe for further study, as the effect of Write Reason on longer-term projects involving the incorporation of a wide range of complex ideas remains unknown, and I would expect that Write Reason’s performance in this setting may exceed that of a typical text editor.
The user study also provided insight into the kinds of strategies pursued by participants when presented with a graph interface to plan an essay. An argument mapping technique based on mindmapping was a commonly adopted style, and users who employed this strategy attained some of the highest scores. Other interesting, more novel strategies were pursued, such as an evidence mapping style. These present some of the possibilities of a free-form graph interface for argument construction - future studies focused on this aspect could provide useful insight into the effectiveness of different structures for representing different kinds of arguments and thinking patterns. Further work could also focus on the degree to which different approaches to graph-based thinking are effective across individuals, or whether they are highly particularised. This is an exciting area of inquiry to be pursued in future work, with the potential for important insight and application to improving the quality of academic reasoning and communication.
12 References


Appendices

A  User manual

To setup Write Reason, install Node.js [nodejs.org/en/download], change your working directory to the one containing the src directory, then run npm install to install required packages. To run the local React server, run npm start. Detailed usage instructions are available in the demonstrational video at https://youtu.be/W5uo2FBUL7c.

B  Fact sheets used in study

The factsheets provided to participants in the study are reproduced below. The factsheets consist of short quotes, edited for clarity and brevity, from the following articles:

https://www.pps.org/article/what-is-shared-space
https://www.hindawi.com/journals/jat/2019/6510396
https://www.pps.org/article/what-is-shared-space
https://www.rbkc.gov.uk/exhibitionroad/sites/default/files/atoms/files/Exhibition_Road_Pedestrian_Behaviour_Study_0.pdf
https://publications.parliament.uk/pa/cm201617/cmwomeq/631/63109.htm#_idTextAnchor070
https://www.science蜣agazine.org/science蜣agazine/05_july_2019/MobilePagedArticle.action?articleId=1502316#articleId1502316
https://www.rsb.org.uk/news/158-biologist/features/1902-focus-on-biohacking#ref1
Please write a response to the question below. Write your response in prose, in the style of a short essay.

QUESTION: Should “shared spaces” in urban planning be promoted?

Urban Design Group

One of the objectives of Shared Schemes is to reduce vehicle speeds to below a level where they are capable of inflicting serious and fatal injuries on vulnerable road users. There is evidence from schemes such as Poynton or Bexleyheath High Street that the approach is successful.

Holmes Report on Shared Space

When asked to rate their experience of a shared space, 63 per cent of the sample rated it as poor, 19 per cent as fair and 18 per cent as good.

Poynton Town Council

The Poynton shared space scheme comprised the removal of all traffic signals, pedestrian guardrailling, and road markings from the junction. In its place a double roundel arrangement of two overlapping circles was outlined using contrasting colours of block paving, with only kerbs of a slight upstand separating the pedestrian areas from the carriageway.

Evaluation of Shared Space to Reduce Traffic Congestion

This research focuses on evaluating and quantifying the traffic congestion relief abilities of shared space designs utilizing Vissim traffic microsimulation software and the economic impact these changes can make. A major pedestrian crossing location on West Virginia University’s Downtown campus along a major urban arterial was chosen as the case location upon which the model was to be built. The results of the analysis show that shared space can reduce vehicle travel time by up to 50% and delays by 66%.

Project for Public Spaces

A shared space design is typically a low-speed environment that desegregates road users and removes traffic controls such as traffic lights, excessive signage and road markings. This “humanization” of the road then frees up space for non-traffic functions. While the level of “sharedness” can vary depending on the interaction between modes of transport, in general, shared space fosters civility and interaction between modes, which enables users to move safely through the space using social cues rather than assigning right-of-way to one mode over another.

Royal Borough of Kensington and Chelsea pedestrian behaviour study

The Royal Borough of Kensington and Chelsea statistics show the average speed of vehicles in the Exhibition Road shared space road in South Kensington is now 27mph – up from 22mph in 2013. Paralympic swimming gold medallist Lord Holmes described this increased speed as “lethal”.

Project for Public Spaces

Where in the late 19th century streets were still part of a coherent scheme, in the 20th century the (re)design of streetscapes became the responsibility of traffic engineers rather than planners or architects. Following the grand vision of freedom promised by the car, the unrestricted flow of traffic became the standard. With higher speeds and more motorized traffic, then, came increased danger on the road, and the resulting demand for safer infrastructure often took the form of a separation of road users. Cobblestones had become concrete or asphalt, while road marking and signs divided the street. Crosswalks highlight
where pedestrians are still able to cross the street, effectively banning them from crossing at any other place.

Holmes Report on Shared Space

Lack of kerbing is a common element of shared space design known as level surface. This causes particular problems in terms of losing the sense of “safe” space. As one parent wrote: “When I was walking with my young children who were taught to walk on a pavement and stop at a curb I almost lost my young daughter who ran into the path of a large car which appeared not to moderate its speed to accommodate the shared space.”

UK Department for Transport

The UK Department for Transport defines shared space as “a design approach that seeks to change the way streets operate by reducing the dominance of motor vehicles, primarily through lower speeds and encouraging drivers to behave more accommodatively towards pedestrians.”

Women and Equalities Committee report

Michael Broderick: “As a person with Cerebral Palsy who walks with crutches and who had previously fallen in a roadway years ago, shared space schemes terrify me. With traditional roadway schemes and traffic lights or zebra crossings I have always had a level of comfort that—despite my disability—I have time to get across the road and that I will be seen. With shared space I have no comfort. I have fear. I am a proud disabled man. I am not a second class citizen. But shared space schemes certainly make me feel like one.”

Dutch Institute for Road Safety Research

Academic traffic engineers at Dutch Institute for Road Safety Research (SWOV) refer to [shared space as] the “creation and design of residential areas,” where “traffic is a guest,” and “the layout should clearly indicate that the primary function of the area is residential.”

Urban Design Group

Everybody uses the streets and spaces in our towns and cities therefore it is essential that a design balances the needs of the very wide range of users. This necessarily requires compromise from everyone. I do not believe that the best design is necessarily one that focuses on the needs of the most vulnerable user group. It is crucial that the design of public realm is not skewed to any particular user group as this will adversely affect others and produce bad design.

Holmes Report on Shared Space

The pattern of non-reporting of accidents in shared spaces to the police seen in our survey, with very few respondents (11 per cent) reporting incidents, calls into question the validity of a) the methodology recommended by the UK Department for Transport on how to monitor operational safety of a shared space and b) previous evaluations of shared space using road accident statistics if this under-reporting has not been taken into account.
Please write a response to the question below. Write your response in prose, in the style of a short essay.

**QUESTION:** Should greater regulatory control be exerted over genetic biohacking?

**Vox**
Biohacker Josiah Zayner has had health problems for years, and some of his biohacking pursuits have been explicit attempts to cure himself. But he’s also motivated in large part by frustration. Like some other biohackers with an anti-establishment streak, he’s irritated by federal officials’ purported slowness in greenlighting all sorts of medical treatments. In the US, it can take 10 years for a new drug to be developed and approved; for people with serious health conditions, that wait time can feel cruelly long. Zayner claims that’s part of why he wants to democratize science and empower people to experiment on themselves.

**Scientific American**
While intending to break new ground in underserved health domains, new forms of participatory health research suffer from a lack of legitimacy. Regulators tend to question the quality and scientific validity of experiments that occur outside of certified clinical trials.

**MIT Technology Review**
Citing the tremendous cost of new drugs, an international group of biohackers say they are creating a knockoff of a million-dollar gene therapy. The drug being copied is Glybera, a gene therapy that was the world’s most expensive drug when it came on the market in Europe in 2015 with a price tag of $1 million per treatment. Glybera was the first gene therapy ever approved to treat an inherited disease. Now a band of independent and amateur biologists say they have engineered a prototype of a simpler, low-cost version of Glybera.

**Science journal**
The US Food and Drug Administration (FDA) has chosen not to formally wield its enforcement power over biohacking, but the agency still has a role in community engagement—education, warning, and standard-setting for activities that pose public health risks. Through its longstanding role in assessing drugs and biological products, FDA is the government regulatory agency equipped with the expertise to assess the safety and effectiveness of genetic biohacking. FDA involvement, therefore, may help to realize the promise of genetic biohacking through guiding biohacking efforts toward interventions that live up to the communities’ hopes.

**MIT Technology Review**
“Biotechnology, more than any other domain, has great potential for human good, but also has the possibility to be misused,” says Daniel Gerstein, a senior policy analyst at RAND and a former under secretary at the US Department of Homeland Defense. “We are worried about people developing some sort of pathogen with robust capabilities, but we are also concerned about the chance of
misutilization. We could have an accident occur with gene editing that is catastrophic, since the genome is the very essence of life.”

Scientific American

A professor of computer engineering, Matt Might, worked with many physicians for multiple years to find even a preliminary diagnosis for his son, who has a rare genetic disease. Matt hunted the genetic mechanisms responsible for Bertrand’s disease with the hope of finding targets for experimental therapies. His subsequent work in building a community of families whose children possessed similar symptoms ultimately aided researchers and physicians in the discovery of a more concrete diagnosis for the rare neurological disorder.

US Food and Drug Administration (FDA)

FDA is aware that gene therapy products intended for self-administration and ‘do it yourself’ kits to produce gene therapies for self-administration are being made available to the public. The sale of these products is against the law. FDA is concerned about the safety risks involved.

Science journal

Apart from legal mechanisms, some biohacking communities have adopted their own ethics restrictions, which, even if not intended to do so, might indirectly avoid harms to public health caused by genetic biohacking. Given that many biohackers who conduct work at home are also members of community labs, their safety policies have the potential to go a long way in promoting safety in genetic biohacking.

The Brookings Institute

Biohacking group DIYbio has provoked outrage among environmental groups. In March 2015, the UN Convention on Biological Diversity released a report discussing the implications of synthetic biology on biodiversity and small-scale agriculture. Earlier this year, the FDA proposed regulations that would require any genetically-engineered organism to go through strict and lengthy regulatory trials to be approved.

Science journal

Experiments to modify genetic expression that once required specialized training and substantial investments in equipment and reagents can now be conducted for a few hundred dollars and with a basic instruction manual. The rise of direct-to-consumer genetic testing has also resulted in individual access to raw genetic data, fueling a variety of health, wellness, ancestry, and relative identification services that offer to interpret those data.

Royal Society of Biology

In 2009, amid concerns that biohackers could create dangerous pathogens, the FBI began sending agents to biohacking conferences and even sponsoring workshops in an attempt to better understand the burgeoning amateur movement. Cooperation with law-enforcement agencies was welcomed by many biohackers, and it was generally concluded that the development of material that could be used in bioterrorism was extremely unlikely.

Science journal

Like government regulation, private governance is important and laudable but not a perfect or comprehensive solution. Private actors may not be inclined to regulate conduct that poses few risks to them, even if safety risks to others are numerous, obvious, and serious. In other cases, the social stigma of violating community norms may simply be an ineffective deterrent.
C  Ethical approval

The ethical approval document is reproduced below:
University Teaching and Research Ethics Committee

11 March 2020

Dear Adam,

Thank you for submitting your ethical application, which was considered by the School of Computer Science Ethics Committee on Wednesday 11th December, where the following documents were reviewed:

1. Ethical Application Form
2. Participant Information Sheet
3. Participant Consent Form
4. Participant Debrief Form
5. Advert
6. Questionnaire

The School of Computer Science Ethics Committee has been delegated to act on behalf of the University Teaching and Research Ethics Committee (UTREC) and has granted this application ethical approval. The particulars relating to the approved project are as follows:

<table>
<thead>
<tr>
<th>Approval Code:</th>
<th>Approved on:</th>
<th>Approval Expiry:</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS14686</td>
<td>07.01.20</td>
<td>07.01.2025</td>
</tr>
</tbody>
</table>

Project Title: Enhancing Text-Editors with Graph Visualisations
Researcher(s): Adam Binks
Supervisor(s): Alice Toniolo and Miguel Nacenta

Approval is awarded for five years. Projects which have not commenced within two years of approval must be re-submitted for review by your School Ethics Committee. If you are unable to complete your research within the five year approval period, you are required to write to your School Ethics Committee Convener to request a discretionary extension of no greater than 6 months or to re-apply if directed to do so, and you should inform your School Ethics Committee when your project reaches completion.

If you make any changes to the project outlined in your approved ethical application form, you should inform your supervisor and seek advice on the ethical implications of those changes from the School Ethics Convener who may advise you to complete and submit an ethical amendment form for review.

Any adverse incident which occurs during the course of conducting your research must be reported immediately to the School Ethics Committee who will advise you on the appropriate action to be taken.

Approval is given on the understanding that you conduct your research as outlined in your application and in compliance with UTREC Guidelines and Policies (http://www.st-andrews.ac.uk/utrec/guidelinespolicies/). You are also advised to ensure that you procure and handle your research data within the provisions of the Data Provision Act 1998 and in accordance with any conditions of funding incumbent upon you.

Yours sincerely

Wendy Boyter
School Ethics Committee Administrator