Argdown and the Stacked Masonry Layout: Two User Interfaces for Non-Expert Users

Christian VOIGT

ITAS, KIT Karlsruhe

Abstract. In this paper I present two user interface concepts: a) Argdown, a simple markup syntax for incorporating argument semantics into online text messages and b) a Stacked Masonry layout for argument maps. Both aim at making the reconstruction and visualisation of arguments easier and more intuitive for non-expert users. These ideas have been implemented in DIO, a new real-time web application.

Keywords. Argument Map, User Interfaces, Visualisation

Non-hierarchic directed graphs are often very confusing to non-experts when used as user interfaces. In this paper two alternatives are introduced.

A) Argdown is an alternative to the 'usual' way of creating argument maps. Most argument map editors[1] have a very similar workflow: the user creates and links up sentences or arguments in separate steps, each divided by mouse interaction. For many users this workflow lacks the speed and fluidity of writing a short and concise comment on a webpage.

The dialogue support tool Arvina [1] offers a more intuitive and less intrusive alternative: users can enter arguments in a simple chat interface. ArguBlogging [2] has another intuitive and simple interface that is similar to comment forms. But even Arvina and ArguBlogging require that the user enters one argument per post. Online messages or posts often contain more than one point, in fact, they often are mini-debates in themselves. Thus, non-expert users should be able to construct arbitrarily complex argumentation structures within a single message. This should not disrupt their normal writing process and remain optional. Finally, these messages should be easy to read, not only for machines, but for other users as well.

Twitter messages[2] or Markdown[3] provide great examples how this can be achieved. Argdown is a Markdown-inspired argument mark-up language. Writing a simple Argdown message is as easy as writing a Twitter message. Let's say user Gregor posts the message 'Argdown is great!' in DIO. This sentence is the twelfth statement that Gregor has posted, so henceforth his statement can be referenced with '@Gregor:12'. Another user could then post a pro reason: '@Gregor:12 +because It's so easy to use.'. Someone else then could add a contra reason: '@Gregor:12 -because it does not support multi-premise arguments.' By using line breaks and plus and minus signs as bullet points, several reasons can be collected in a pro and contra list by a single user.

By indenting such list items we arrive at tree structures. And by referring to statements by titles we can even define multiple outgoing and incoming relations of the same statement within the same message.

A first version of Argdown is implemented as a context-free grammar and parser.[4] This parser is used in DIO for live code highlighting during typing and for parsing the message after it is posted.

B) The Stacked Masonry layout is an alternative to visualizing argument maps as directed graphs. The latter have several disadvantages: if the dialectical relationships are complex, the text in the graph has no clear reading order and the user gets lost. Even if the software uses sophisticated layout algorithms[5], in complex maps a tangle of arrows is inevitable and too much horizontal screen space is used. Alternatives that are already in use are a simple two-column list[6] or a tree structure[7] of pro & contra reasons. Both limit the dialectical structures that can be visualised and use up too much vertical screen space.

The Masonry layout[8] uses screen space in an optimal way. It minimizes gaps between elements by putting them in a flexible grid. The Stacked Masonry layout combines several such grids in a 'Russian doll' style, so that each element can contain other sub-elements within its own Masonry grid. These containments visualise the pro and contra relations. Multiple outgoing relations can only be visualised for the largest element, but each element can be maximised on a separate Stacked Masonry page. All transitions are animated in order to preserve continuity.

While both Argdown and the Stacked Masonry layout may have important advantages over conventional methods of editing and visualising argument maps, they could still prove to be too demanding for many non-expert users. The claims I have made about the advantages and disadvantages of different kinds of user interfaces lack solid empirical evidence even though they are based on various experiences with argument maps. It would certainly be an important and interesting task to test their validity.[9] Here I have only used them as plausible working hypotheses that explain the motivation and inspiration behind DIO. In my opinion, the results at least prove the fruitfulness of these hypotheses.

References

[1] Lawrence, J., Bex, F. & Reed, C. (2012) "Dialogues on the Argument Web: Mixed Initiative Argumentation with Arvina" in *Proceedings of the 4th International Conference on Computational Models of Argument* (COMMA 2012), IOS Press, Vienna.

[2] Snaith, M., Bex, F., Lawrence, J. & Reed, C. (2012) "Implementing ArguBlogging" in *Proceedings of the 4th International Conference on Computational Models of Argument* (COMMA 2012), IOS Press, Vienna.

[3] Benn, Neil, and Ann Macintosh. Investigating Visual Languages for Argument Mapping. In *Working Notes of the Narrative and Hypertext Workshop* at the 22nd ACM Conference on Hypertext and Hypermedia, 2011. Eindhoven.

---

[4] https://github.com/christianvoigt/argdown

[5] i.e. www.yworks.com

[6] www.debate.org, www.procon.org, www.debatepedia.idebate.org

[7] One of several alternative visualisations at www.debategraph.org and www.evidence-hub.net

[8] http://masonry.desandro.com/, one of the first sites that used this layout was www.pinterest.com

[9] A similar project is presented in [3].