

Design Guidelines

by Hai Tran



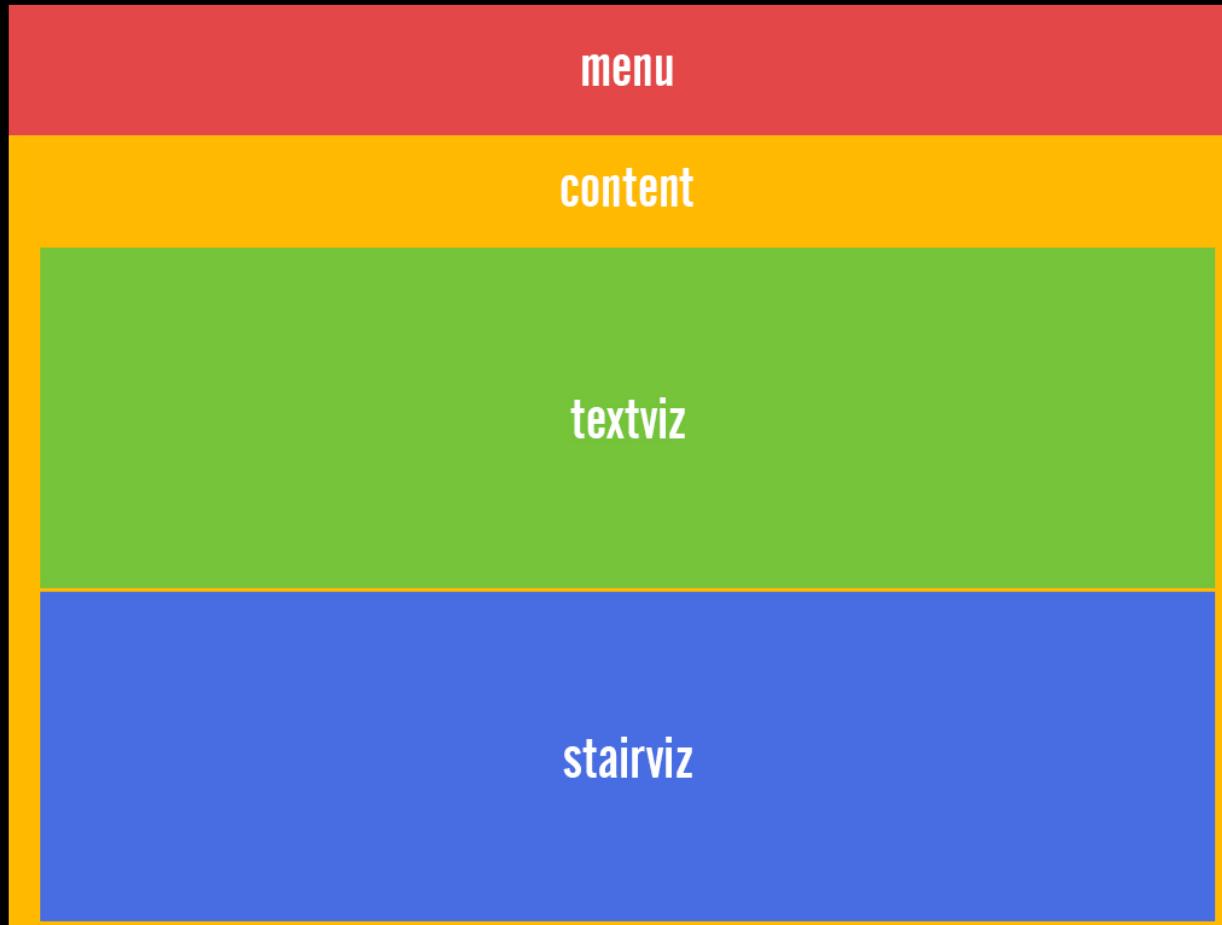
This is our screen.



I'm drawing it at 1024 x 768, but the size shouldn't matter much.



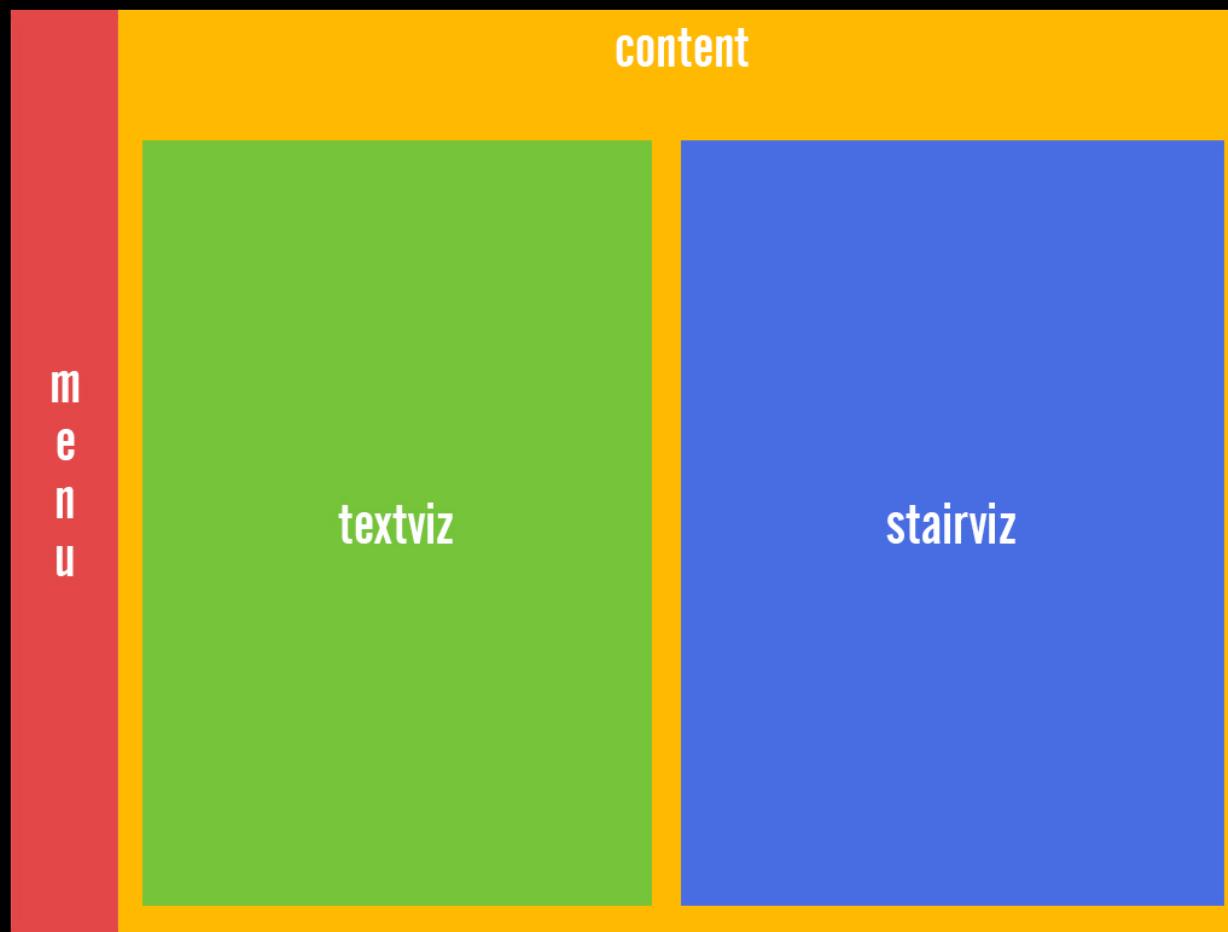
Our page will be divided into 2 main components: **menu** and **content**. **content** is again divided into 2 smaller components: **textviz** and **stairviz**. Size and position of the components should be defined using CSS. And these values should be responsive, not fixed.
If we follow this rule, we can organize them like this...



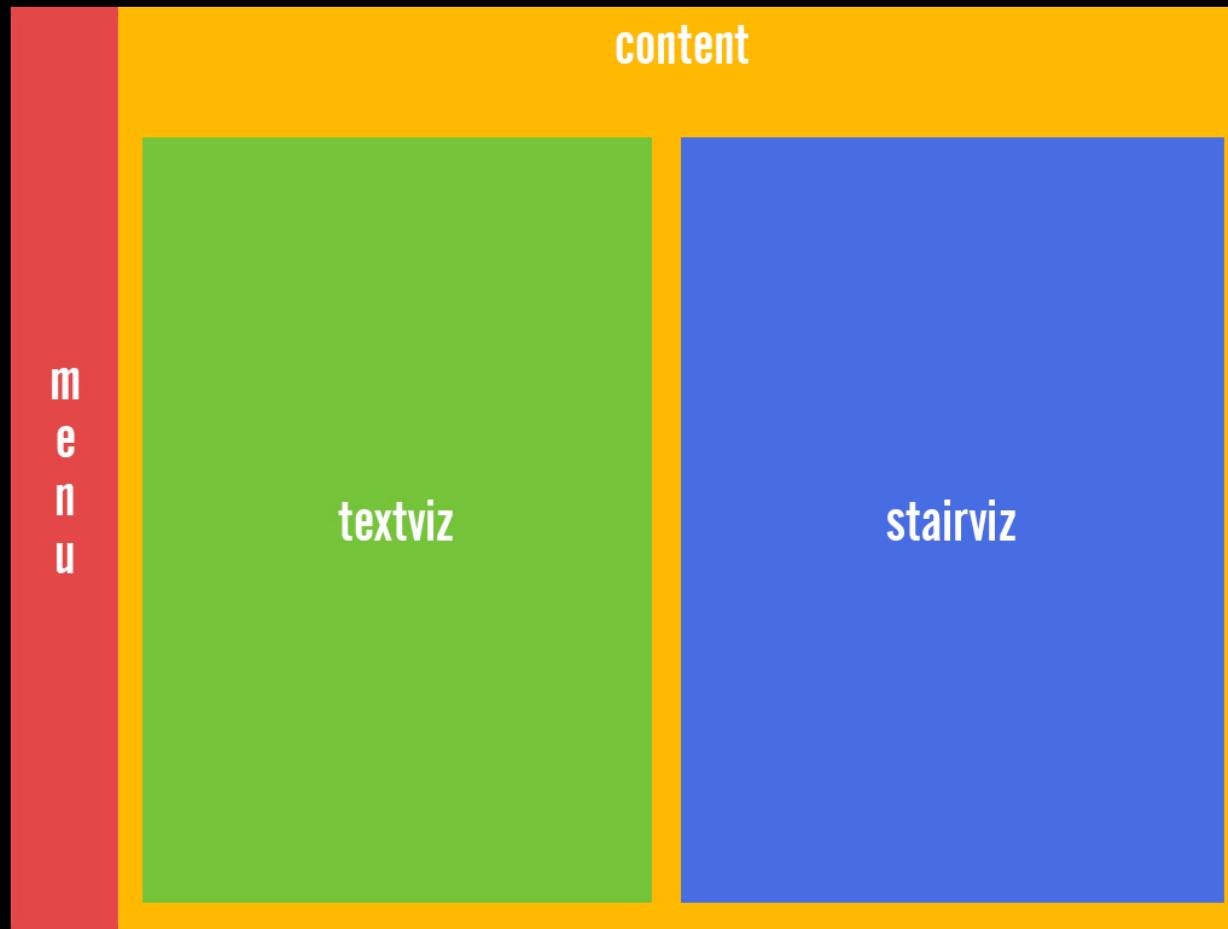
... or like this.

CSS helps us make this immediate switch by editing only a few lines of code.

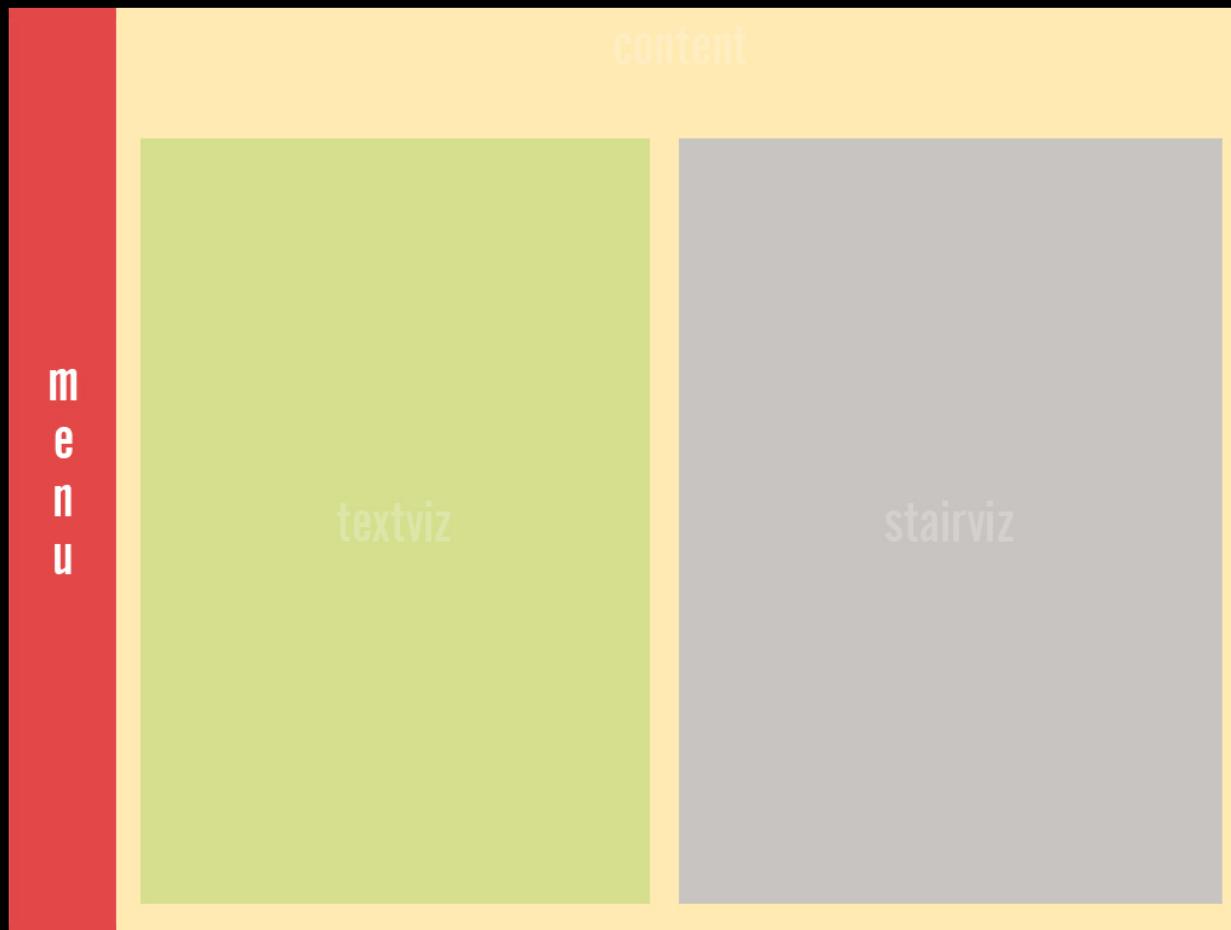
Since I don't like looking at paragraphs that have width much larger than height,
I'll stick with this second layout in this guidelines. If I have time, I'll add drawings
for the first layout.



Now let's go into details and see what each container has.



1. Menu



Our menu will have background color of #333333.

Size: height 100%, width 10%.

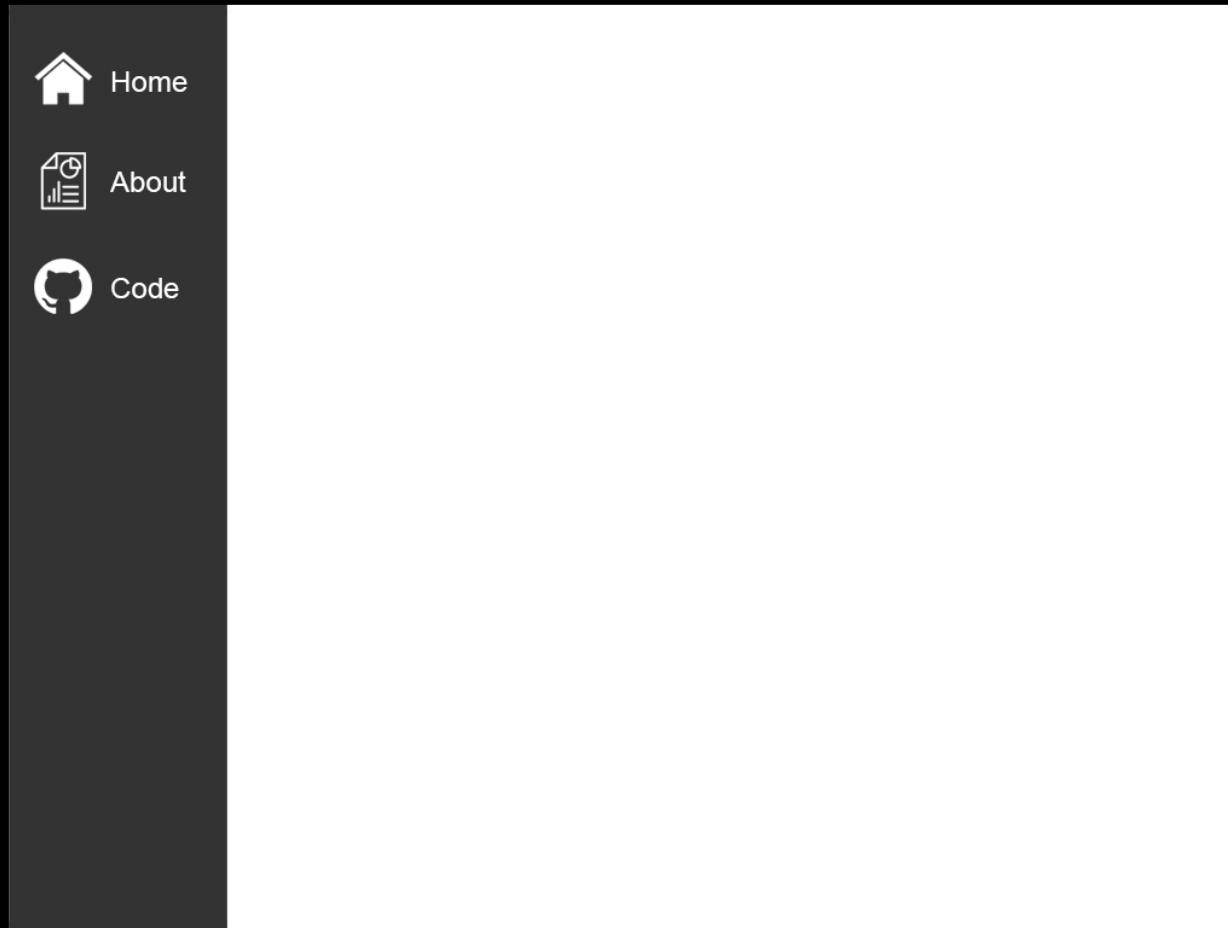
Note: these are height and width of the entire webpage.



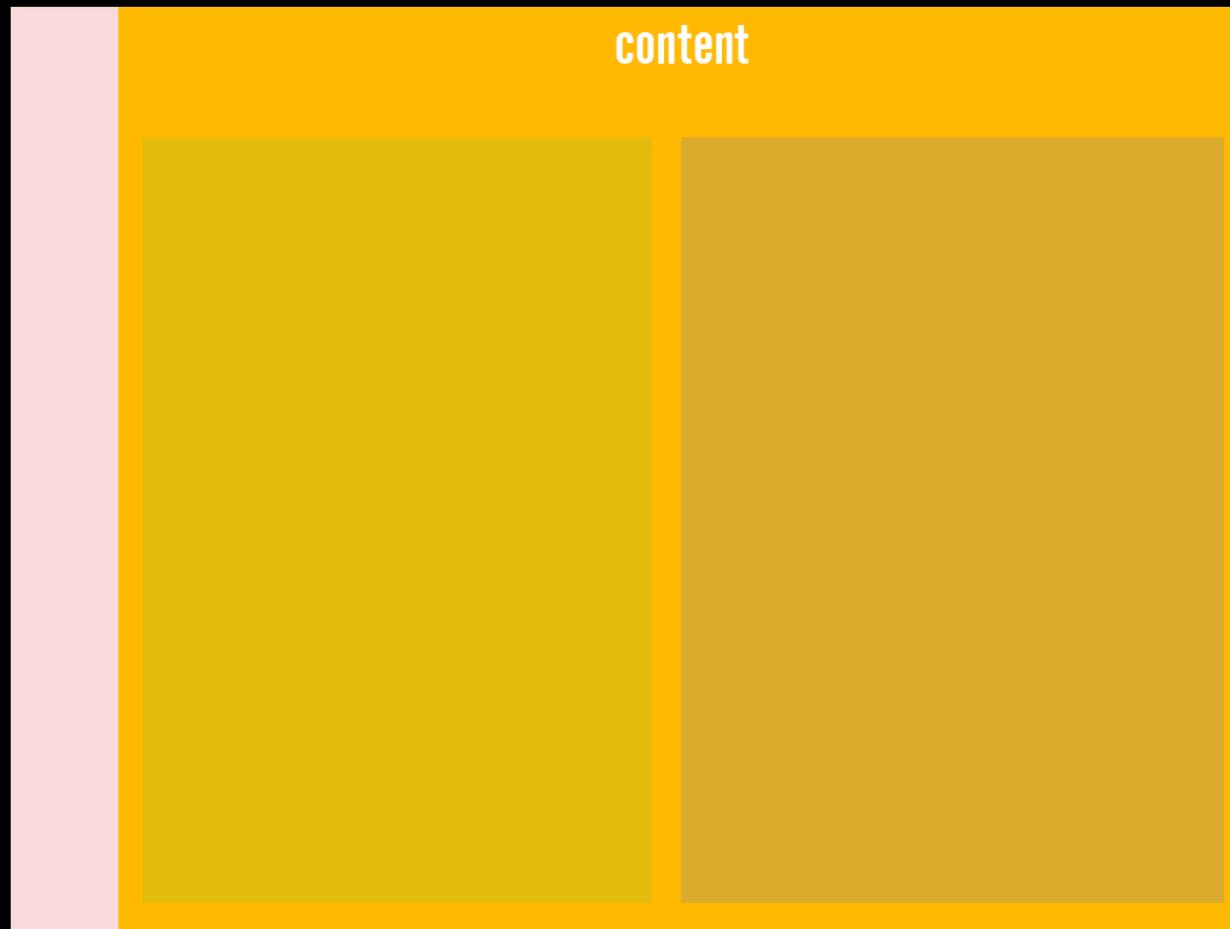
menu doesn't contain many things except icons. Each icon links to a webpage. Unless the icons are hand-drawn using d3, their maximum size should be 48px * 48px. Each has a top-margin of 40px and is centered horizontally within its container.



On mouseover, menu will expand to display texts that go with the icons. Onmouseleave, menu will return to its original state. Whoever does this will be responsible to choose an appropriate width for the expanded menu.



2. Content



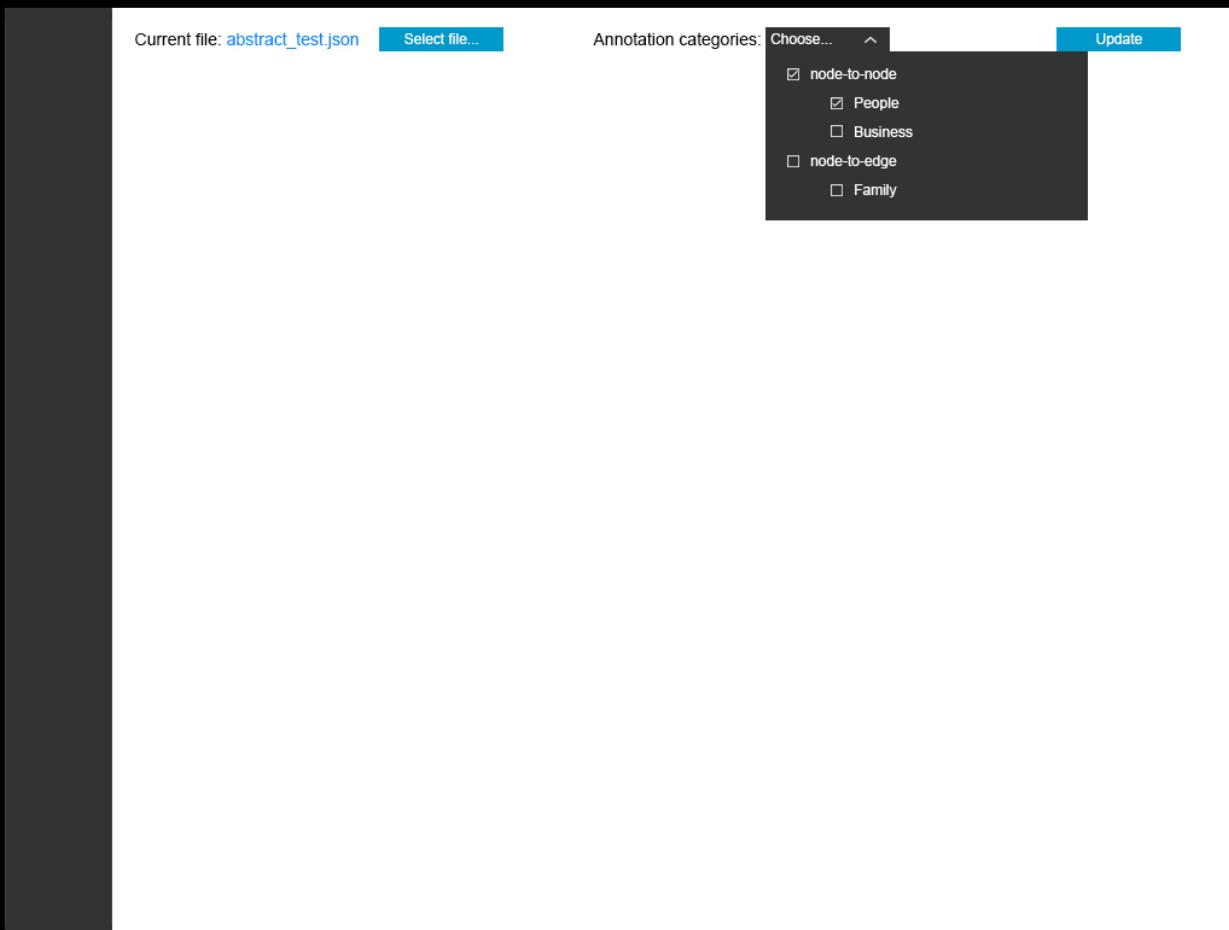
CSS properties of content:

- background color: white
- width: 90%
- height: 100%
- padding (all sides): 10%

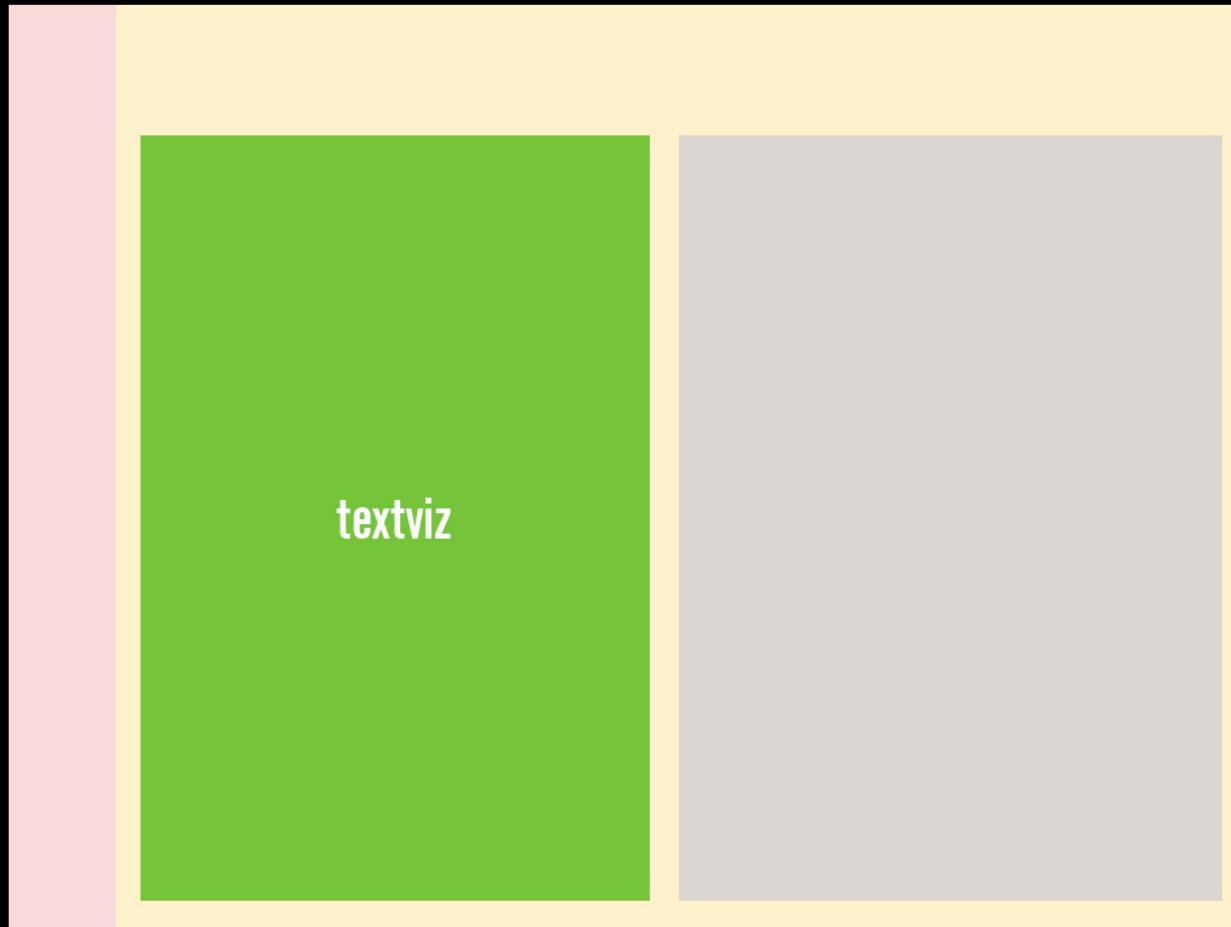


The top of content looks like this.

- Color of buttons is `#0099cc`
- The dropdown list may not be the default dropdown provided by HTML. This can be a hidden div that only appears on mouseclick.
- The list should be populated directly from data file (annotation file).

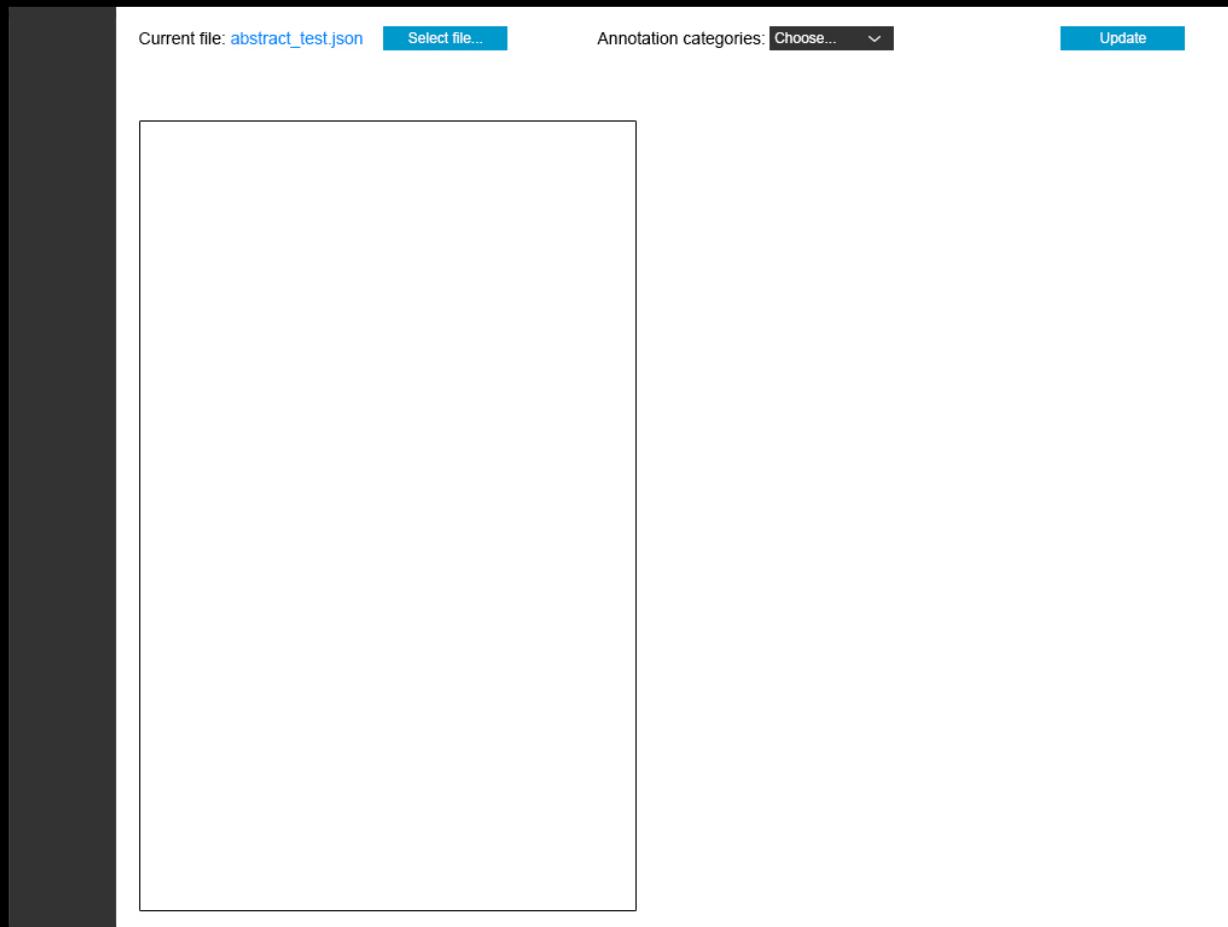


3. Textviz



CSS properties:

- height: 85%
- width: 40%
- background-color: white
- border: black
- margin (all sides): 2%



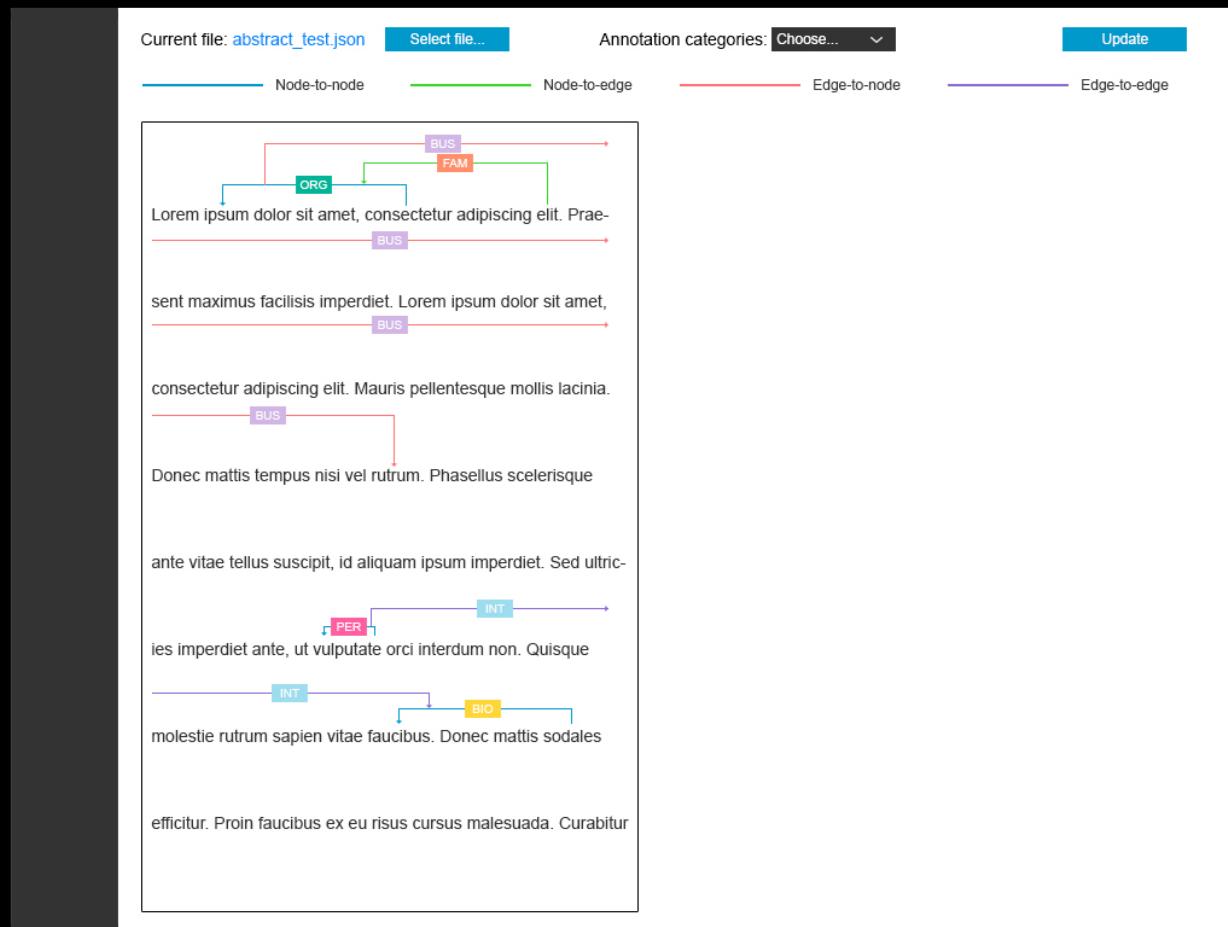
This textviz is used for displaying document in form of paragraphs, similar to how BRAT and ODIN do.

Connections (edge/link/arrows) will be color-coded based on their type:

node2node = #0098ce, node2edge = #3ad531,

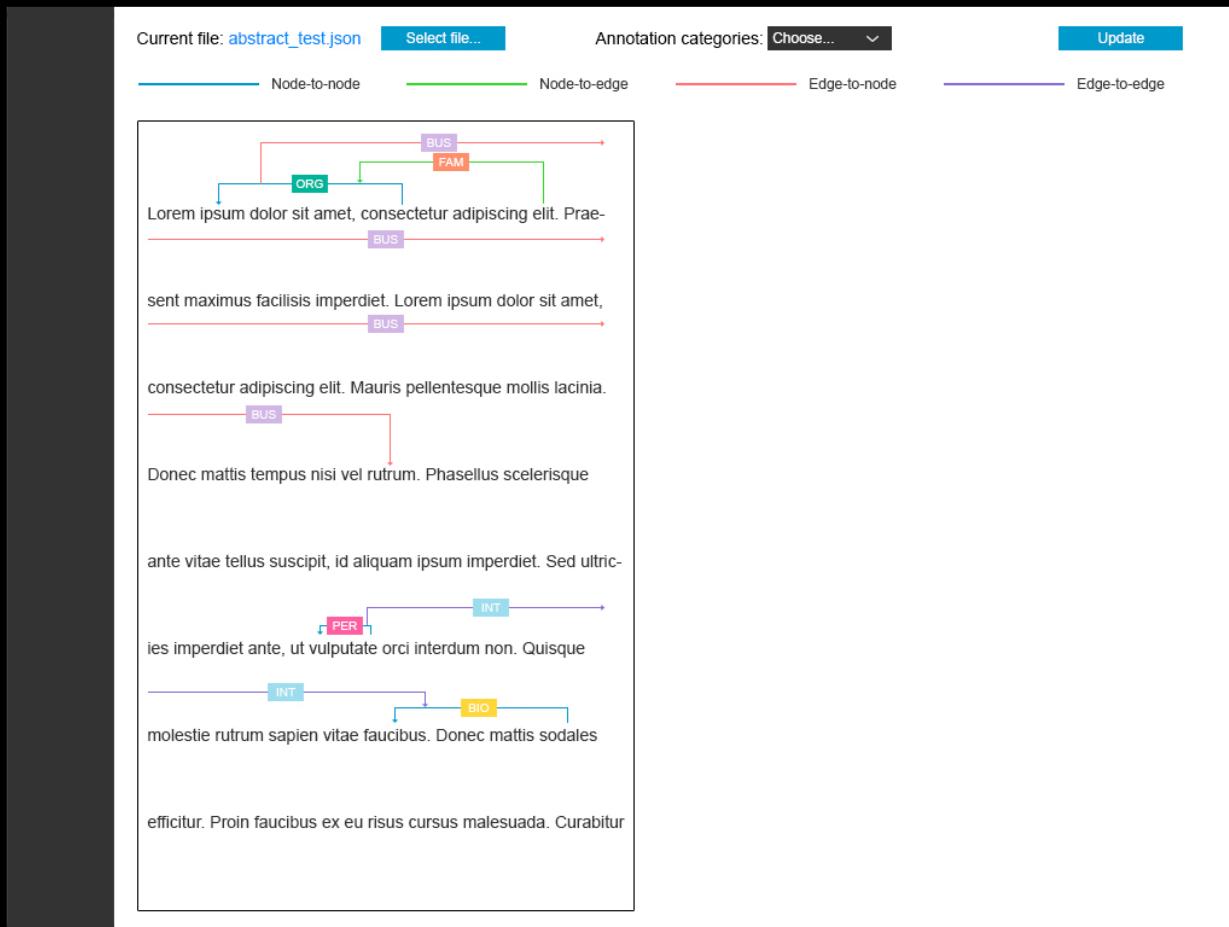
edge2node = #ff7376, edge2edge = #8a6ad4

(Please check **Readme.txt** in folder **data** for details)



The labels are colored randomly, but still need to check for similarity (search for “Detect similar colours from hex values” on StackOverflow).

We need an algorithm to keep labels from overlapping each other. On next page, I'll list a few rules that I've come up with.



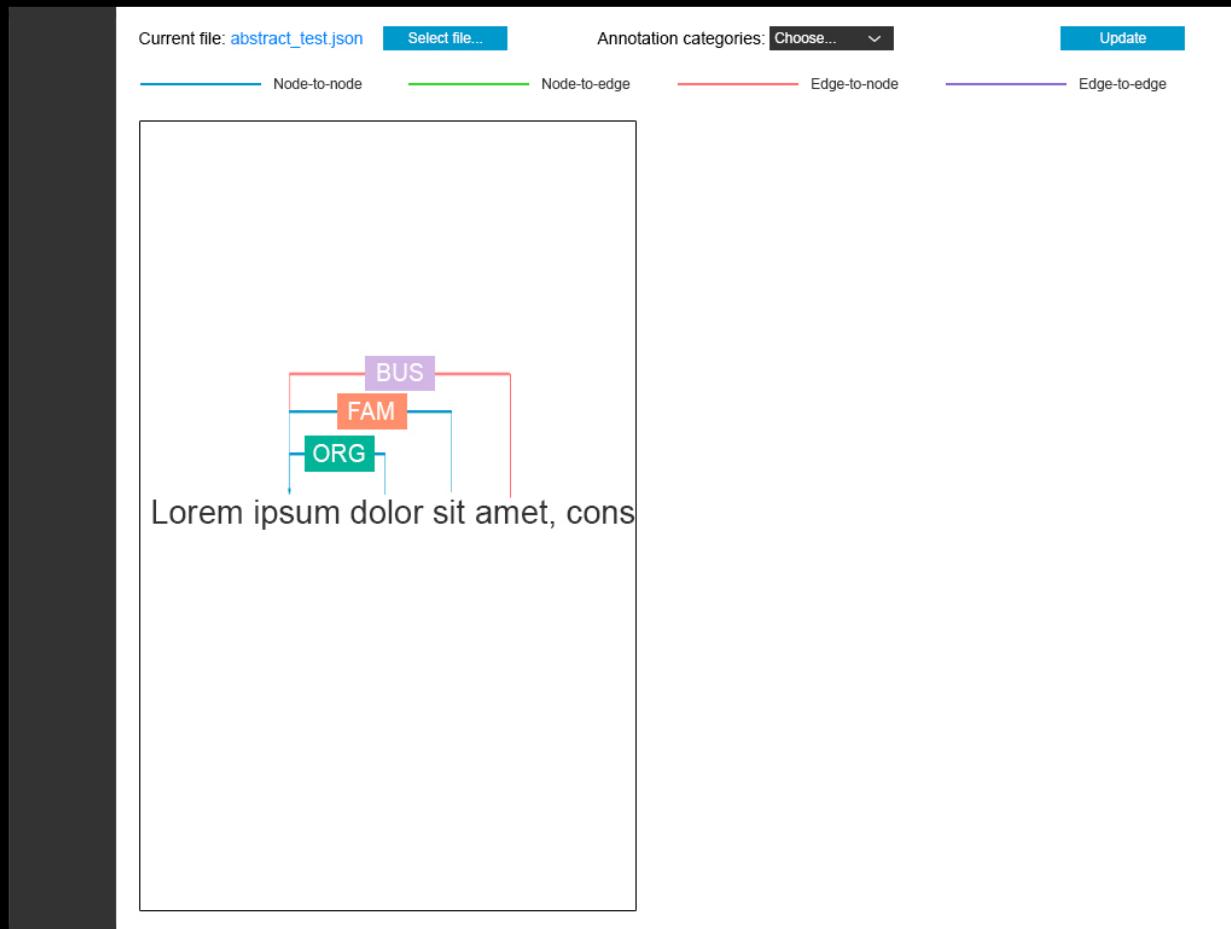
1. There maybe more than 1 arrows point to the same node/word. If it happens, we stack inward arrows together and outward arrows together.
While we stack arrows that go in/out of a word, we also separate them by their directions (left/right).

Because of this, each word will have 4 stacks at most:

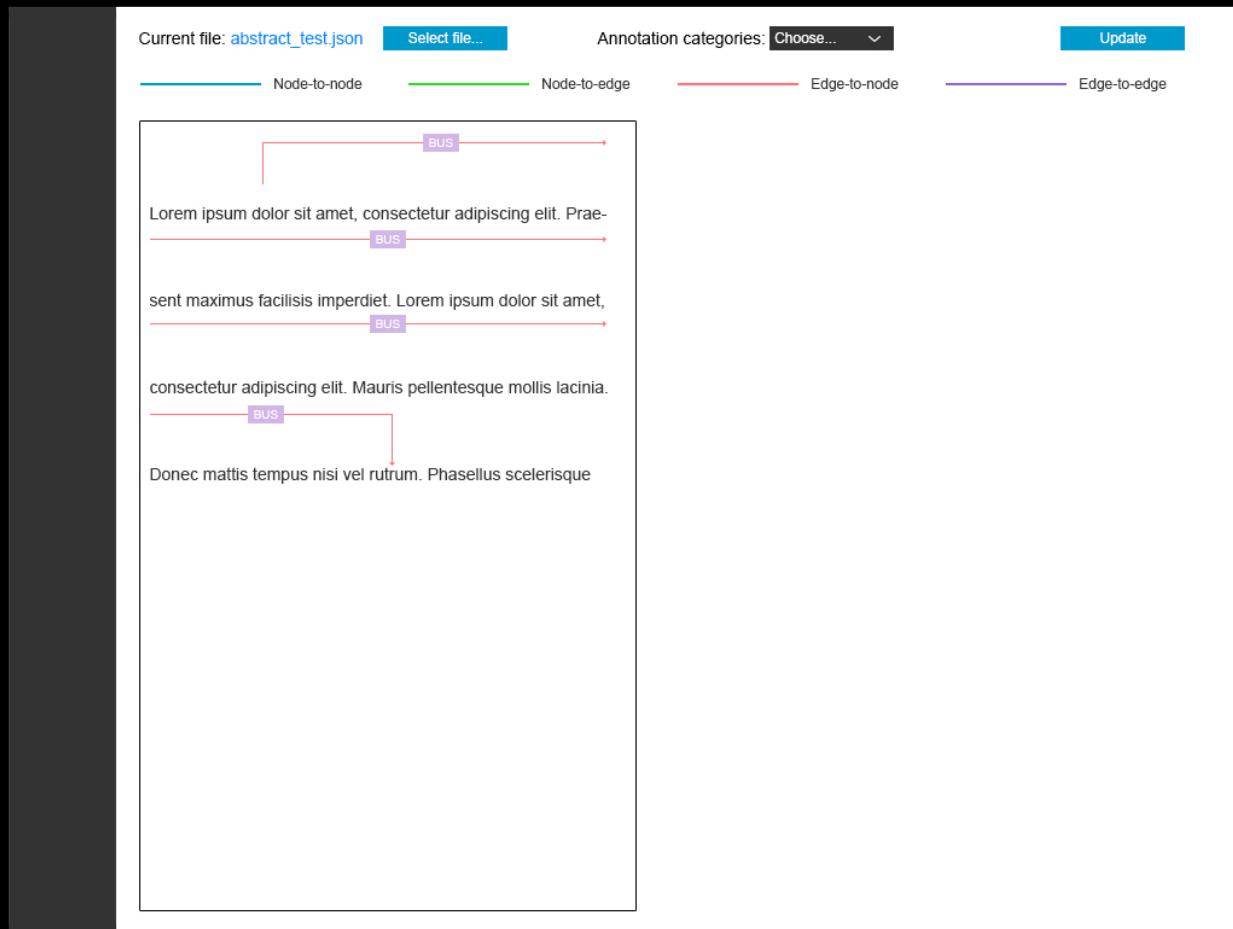
- In + left
- Out + left
- In + right
- Out + right



2. Height of an arrow in a stack is determined by how far it travels: the shorter the distance it travels, the shorter its height.



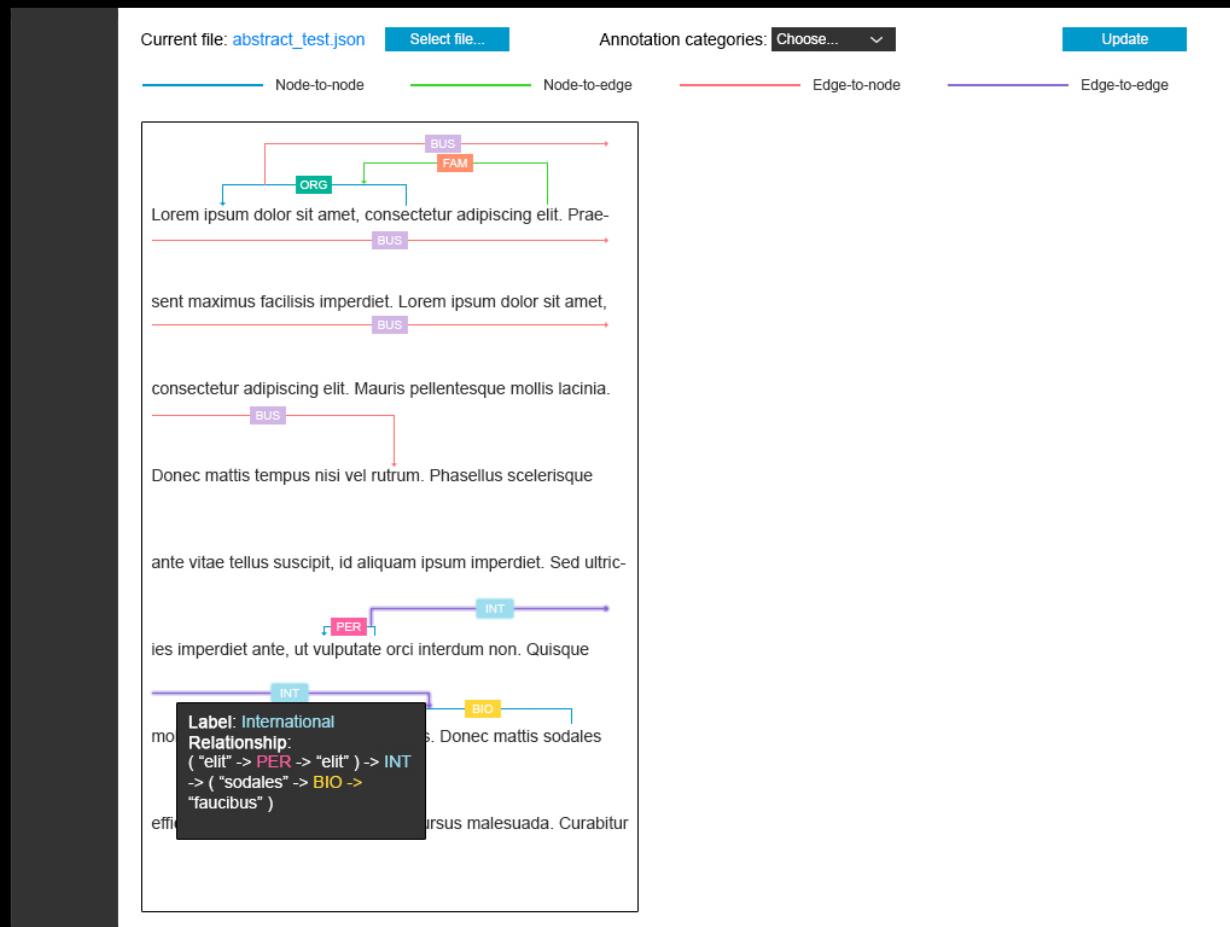
3. If an arrow spans over multiple lines, replicate its label on all lines.
On each line, put the label at center of the arrow segment.



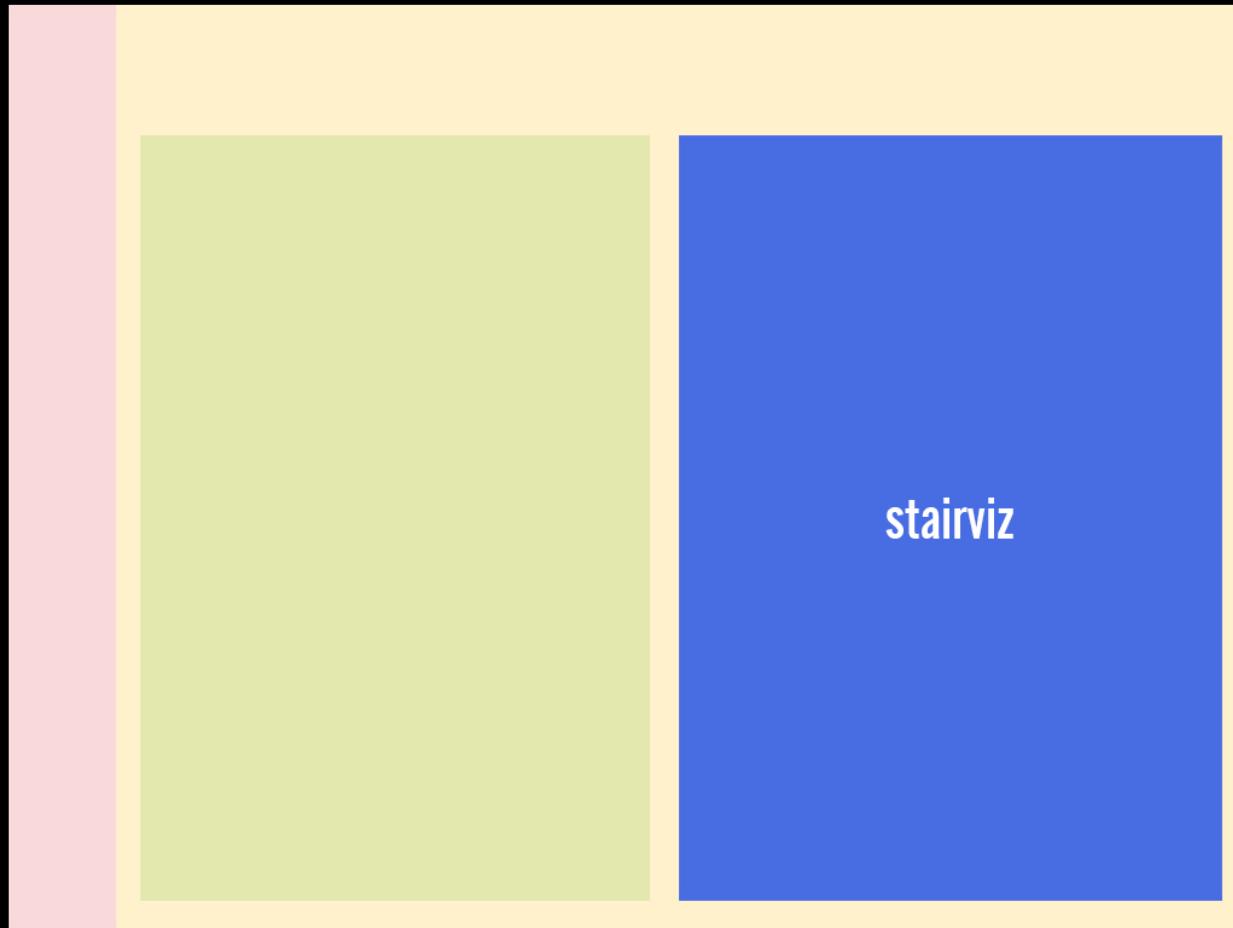
When user hovers on an arrow, it will be highlighted.

When user hovers on a label, details of that label will be displayed.

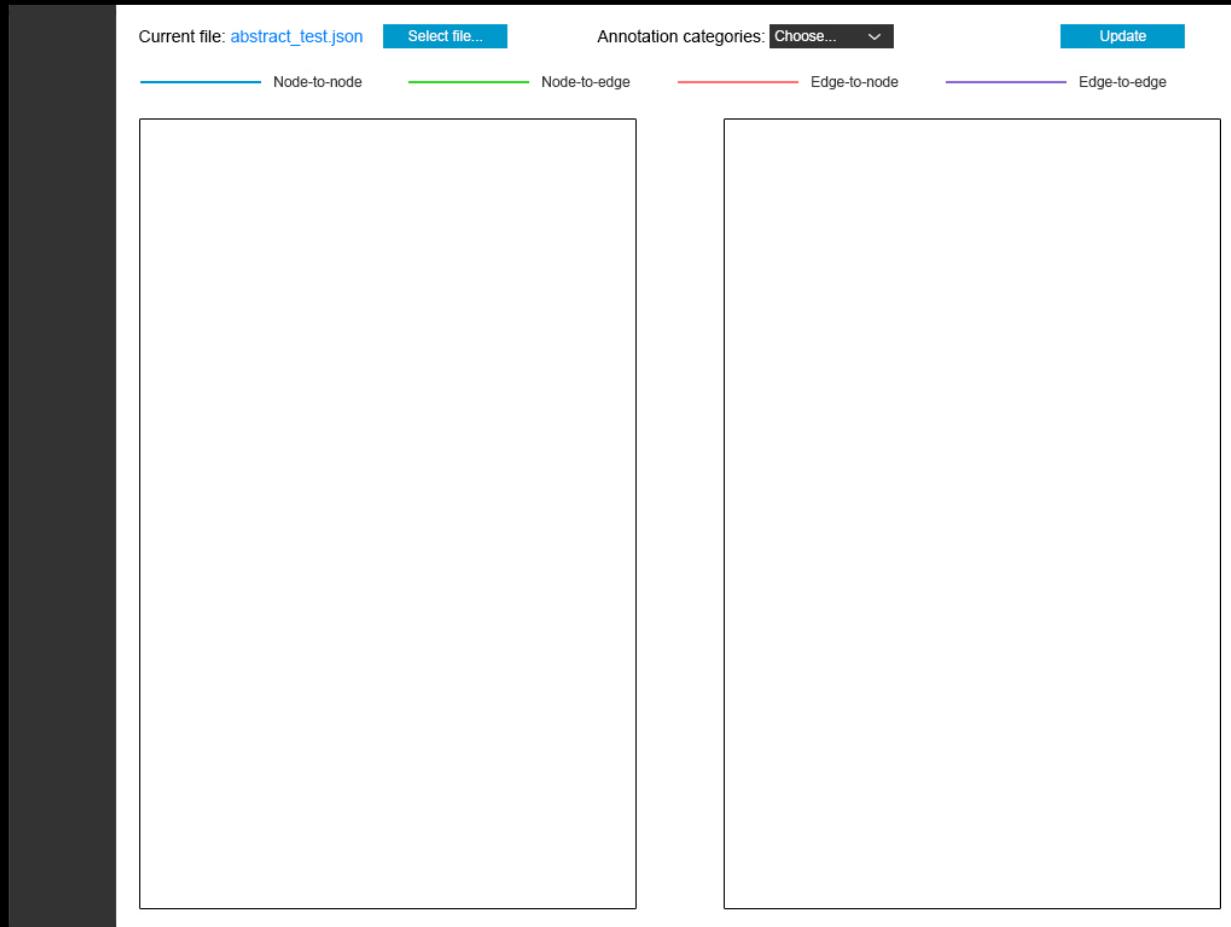
That's our first viz. Now I'll describe the second viz.



4. Stairviz



CSS properties: exactly the same as textviz

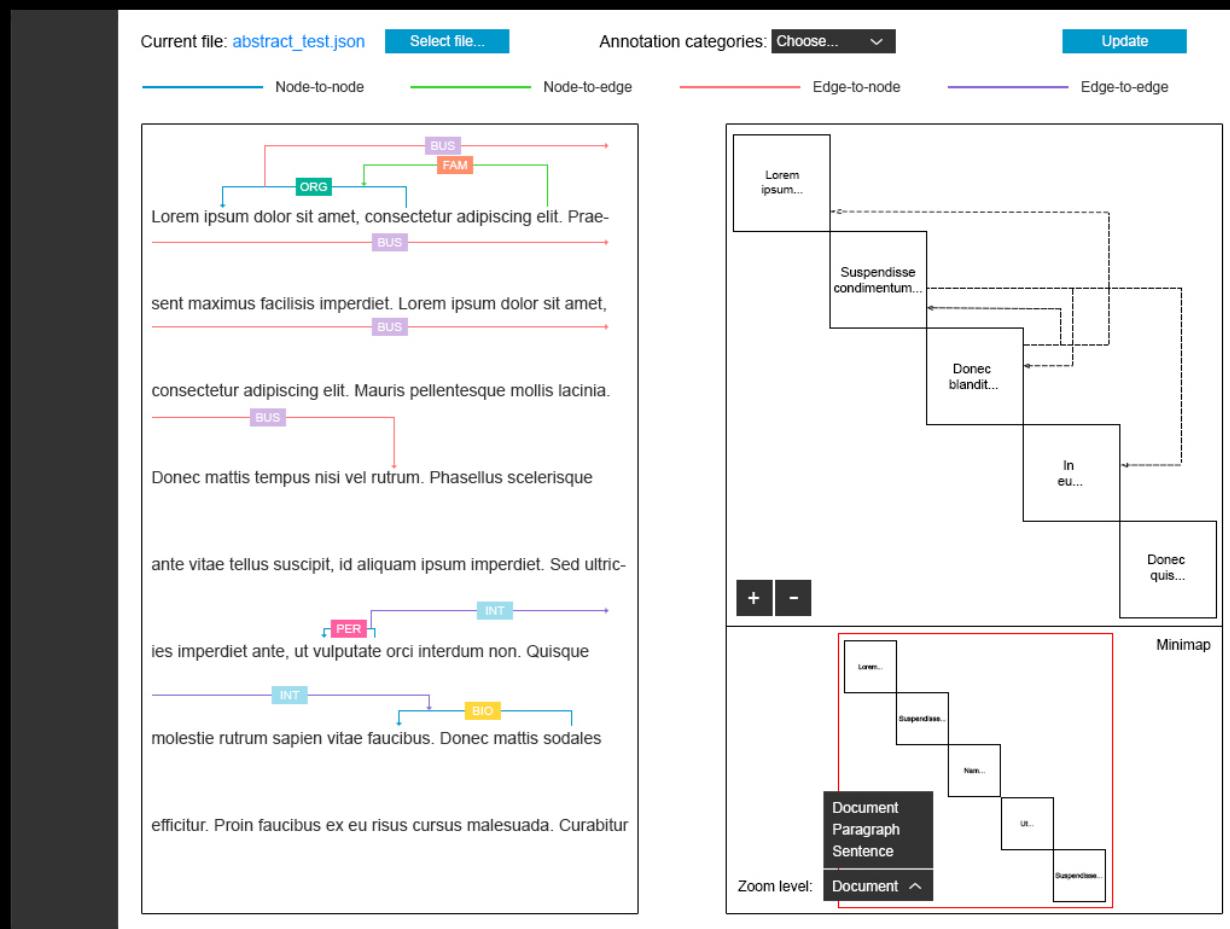


[Stairviz](#) consists of a “stair” chart on top and a minimap at the bottom. The minimap shows our current position in the entire document / a paragraph / a sentence (determined by Zoom level).

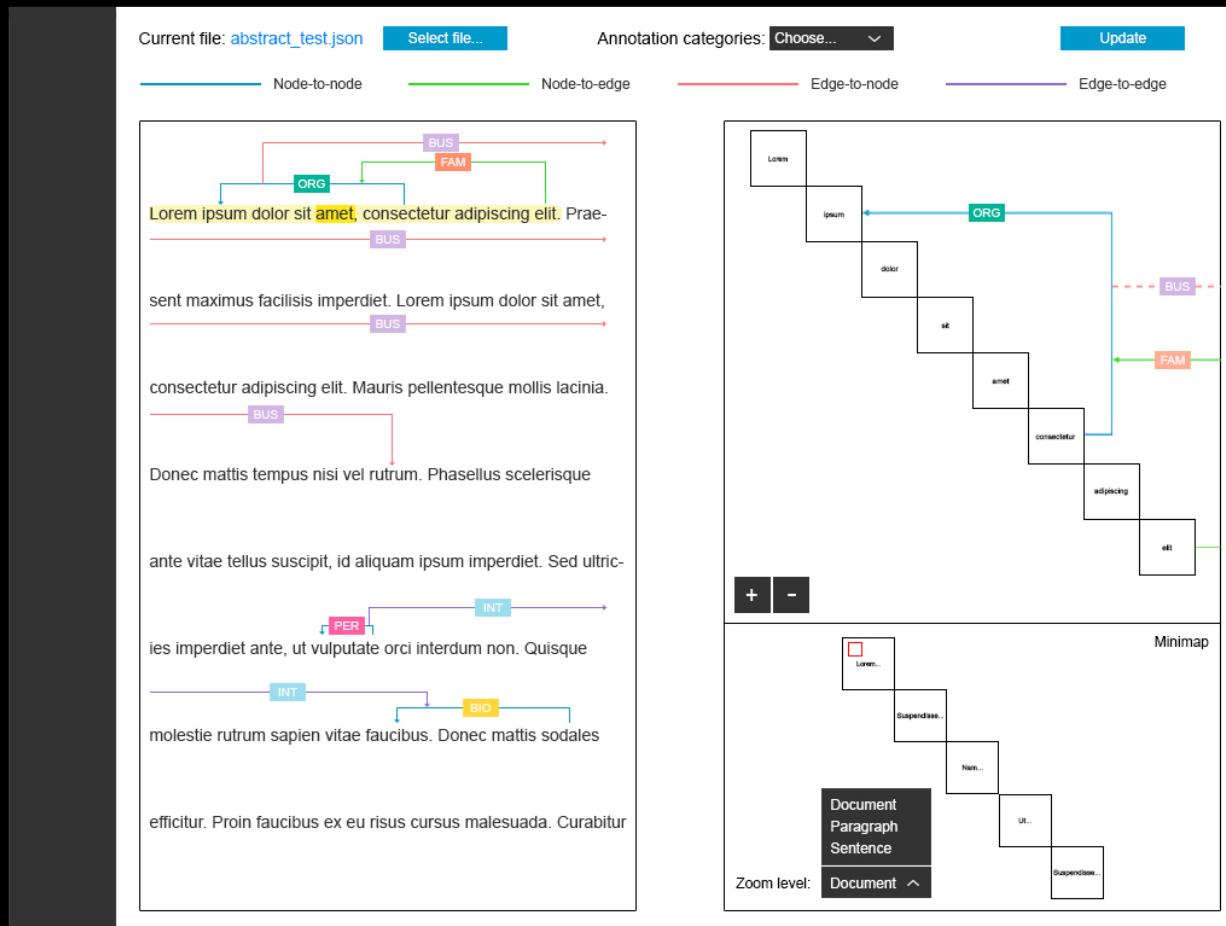
If Zoom level = Document, the minimap shows blocks of paragraphs.

If Zoom level = Paragraph, the minimap shows blocks of sentences in a selected paragraph.

If Zoom level = Sentence, the minimap shows blocks of words in a selected sentence.

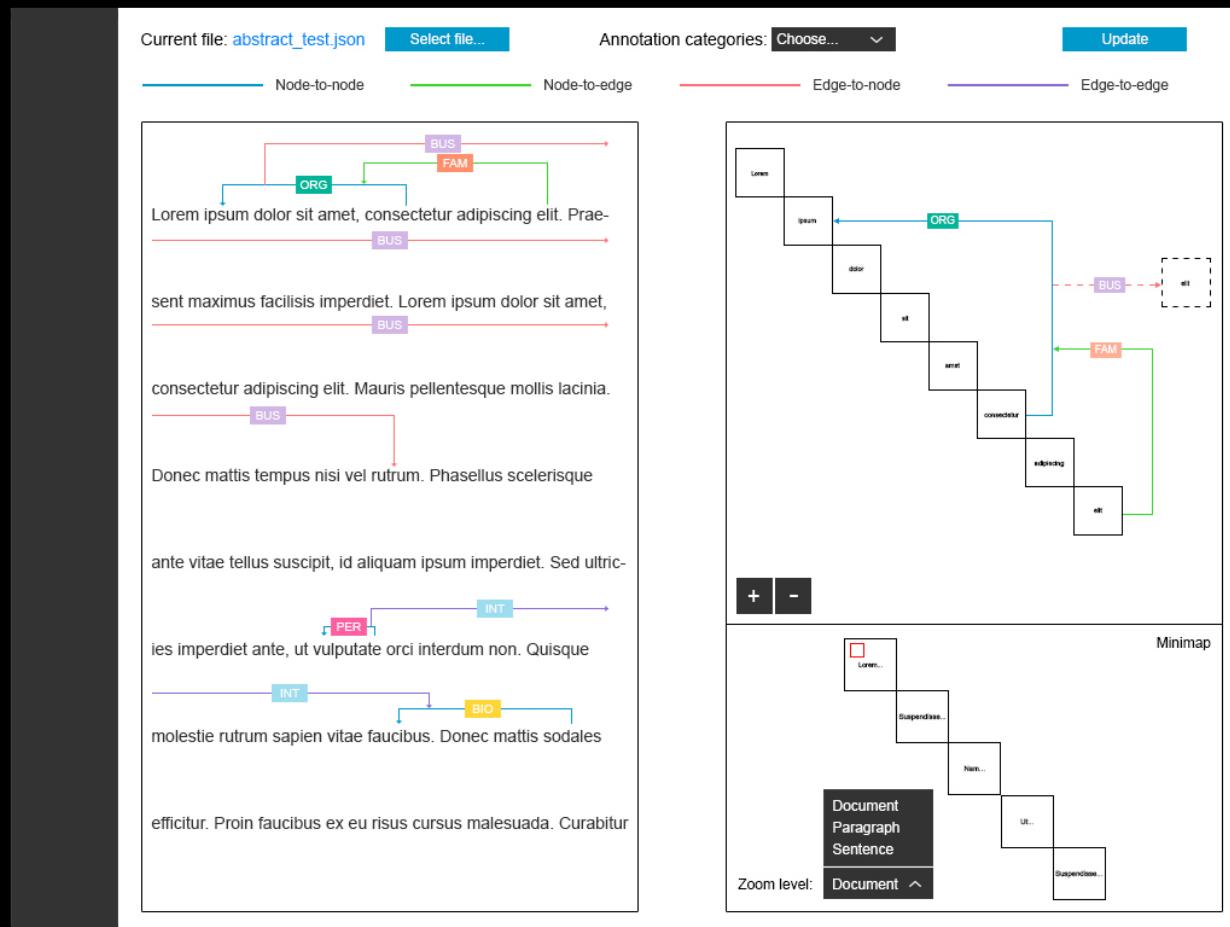


When user hovers on a word in **textviz**, that word is highlighted.
When user clicks on that word, the entire sentence is highlighted. That sentence will then be displayed in **stairviz**.

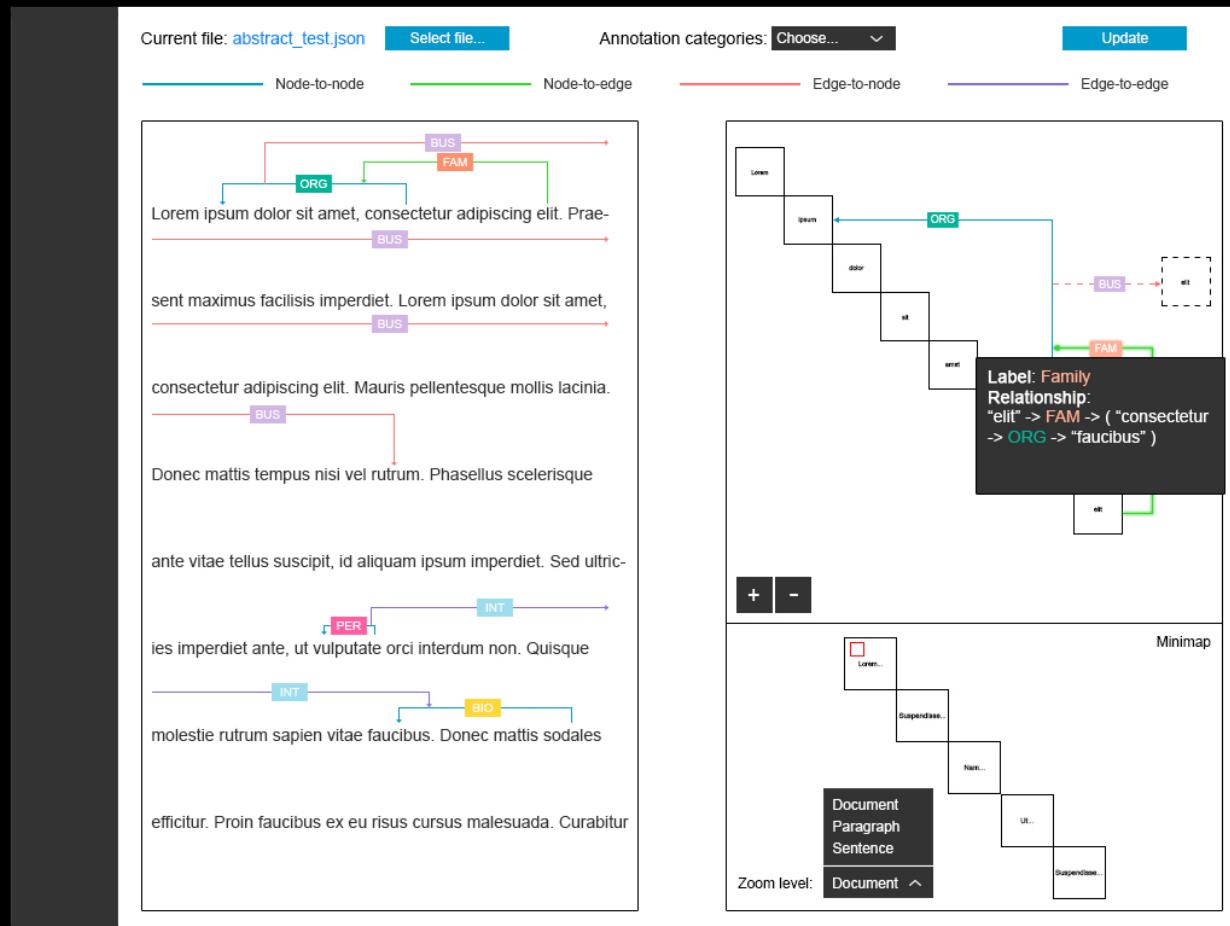


User can zoom manually using +/- buttons right above the minimap and also use mouse to move the main chart around.

Connections that link to another part of the document are represented using dotted lines. User can click on the dotted line to see the other end of the arrow.

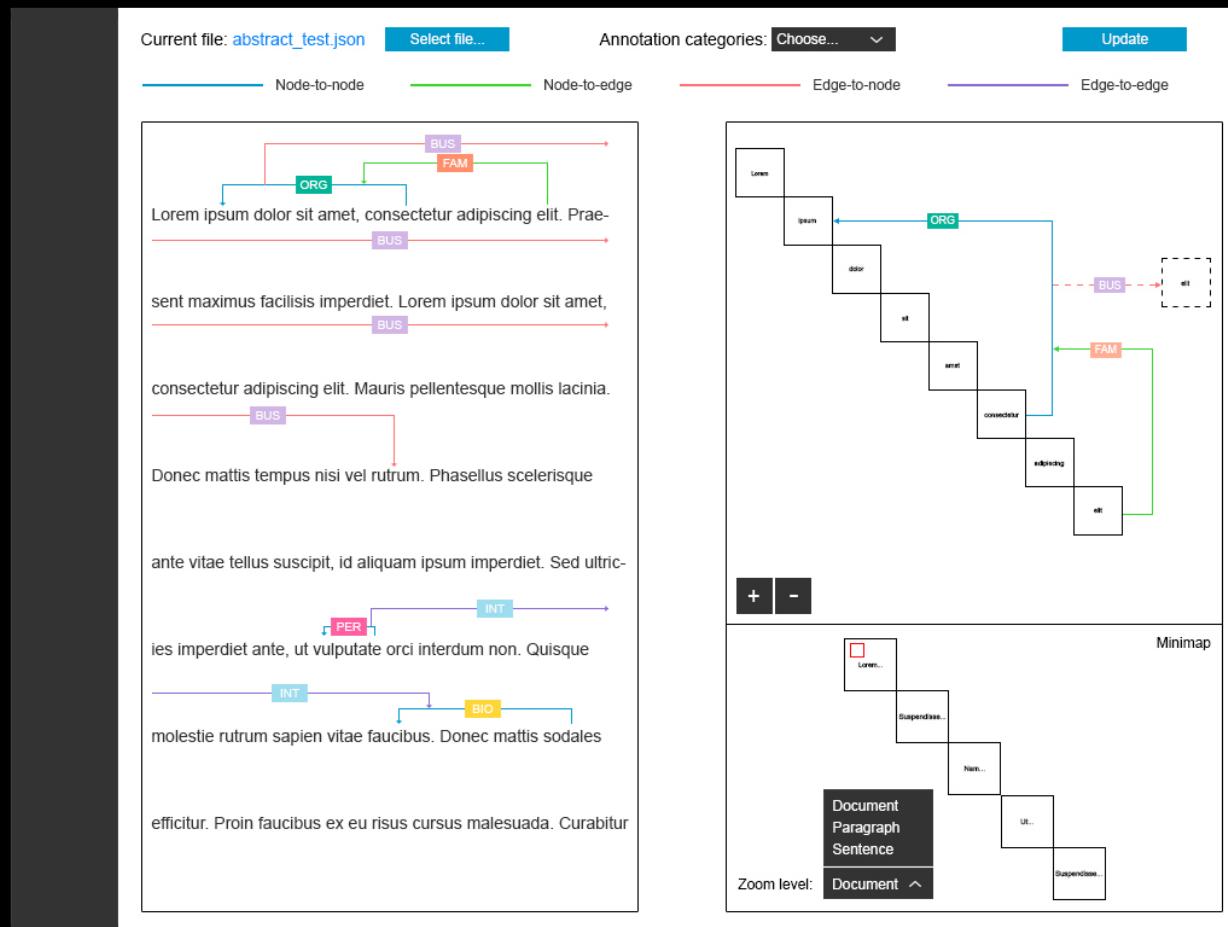


When user hovers on an edge, the edge will be highlighted.
When user hovers on a label, its details will be displayed.



Rules:

1. Minimap does not show any edge.
2. Every visible edge (in the main chart) should consist one vertical line on the right and two horizontal lines.
3. Label should be put at the center of the top horizontal line.
4. All connections point from/to an edge start/end at the vertical line of the edge.
5. Similar to the rules I've defined for textviz, the edges in this viz are stacked and ordered by their directions (in/out, left/right) and how far they travel.



And... Yeah, those are what I have had so far.
The design guidelines end here.

