Idea Tracer

Rob Thompson
robthomp@uw.edu

Lucy Williams
lgw2@uw.edu

Sam Wilson
samw11@uw.edu

ABSTRACT
The goal of this project was to create an interactive visualization to explore segments of matching texts between U.S. Congressional bills, in collaboration with political science researchers. We focused on matchings between the Patient Protection and Affordable Care Act, and created a visualization based on a three-layer approach. The first layer showed an overview of matchings across the whole bill, the second a zoomed-in version of the first, and the third, more information about specific sections of the bill and its matchings. In the third layer, we show matching text with highlighting to indicate matchings and differences. Based on feedback from our collaborators, we believe that we have created an effective tool for exploring the data set.

Author Keywords
PPACA, Obamacare, Law, Bill, Section, US Congress, Political Science, Idea Tracer.

INTRODUCTION
We have designed and implemented an interactive data visualization to support novel computational political science research, which seeks to trace ideas through successive pieces of US legislation. In order to find such trails of ideas, researchers John Wilkerson and Nick Stramp identified areas of reused text between U.S. congressional bills using the Smith-Waterman local sequence alignment algorithm. Specifically, they compared each section of all congressional bills introduced in 111th Congress with each other section. Each bill can contain hundreds of sections. Their database contains the results of 1.8 million text comparisons. Not all instances of aligning text contain reused policy ideas, however. They sought a tool that would allow them and other researchers to easily view aligning pieces of text in a visual display and that would give access to a large amount of the data that they had associated with each alignment.

Our final design centers on only one bill at a time, as suggested by John and Nick at the beginning of the project. We used the Patient Protection and Affordable Care Act (PPACA) as the focus of our visualization throughout our design process and in the presentation of our final design, though any other bill in their database could be substituted. After applying a threshold to discard weak matchings, we had about 1,500 comparisons to present. We used a three-layer approach to allow users to see an overview of the distribution of alignments by political party over the whole bill, then drill down to see alignments for specific sections of interest, and finally select specific alignments to see aligning texts side-by-side, along with more information about the bill from which the aligning text comes. Aligning text and differences are highlighted to facilitate comparison.

Our visualization gives researchers like John and Nick easy access to their data while also giving non-researchers a sense of the history of a bill's text. The overview layers point users to sections of interest, either through prior knowledge of the featured bill ("I'd like to investigate all alignments with section 2303 of the PPACA which deals with family planning services") or by the context provided by the overview ("Nearly all the alignments from that section are from Republican-sponsored bills - I wonder what that text says and how closely the ideas in those alignments match"). Once an interesting section is chosen for exploration, alignments are displayed semi-chronologically on a chart, represented as vertical bars positioned according to the position of the alignment within the text of the featured bill, and colored and filled to show party and congressional group of sponsor, respectively. This chart not only directs users to specific alignments of interest, but could also point to the very "idea traces" that this project seeks. Finally, the placement of text side-by-side with matching and different text highlighted in different colors allows researchers to read each alignment without relying on memory to know what was the same and what was different.

RELATED WORK
We were unable to find work that closely related to our project, namely a visualization of a large amount of closely related text comparison data. John Wilkerson and Nick Stramp had created a handful of static visualizations [4] before enlisting us that we took as a base design and we drew further inspiration from a number of scattered sources including version control GUI's, text alignment charts, and a graph from XKCD[2]. Though we didn't consult this area of research while developing our design, a promising area for related work is interactive exploratory visualization of gene sequencing alignment data, such as VistaChorm [3].

METHODS AND DESIGN

Data

Our data comes entirely from Wilkerson and Stramp. The database contains approximately 10,000 bills, 200,000 sections and the results of 1.8 million comparisons. At John Wilkerson's suggestion we focused on comparisons that involved the final version of the PPACA, the Affordable care act, of which there are approximately 20,000. We applied some basic thresholds to eliminate non-matches and ended up with roughly 1,500 comparisons with any weight.

Each comparison includes the alignment score, the number of gaps, the number of matches, matched string from both documents, and the start and end locations of matched strings from both documents. We also had access to the full, formatted text of each bill section and meta information about each bill including when it was introduced, whether it was introduced to the house or senate, if and when it became law, the bill sponsor, and the political party of the bill sponsor.

Overall Approach

We initially experimented with a adjacency matrix type design. This is also the approach John Wilkerson suggested. We found the matrix to be too large and too sparse to be tenable however. Since we were highlighting comparisons around one bill in particular, we decided to design a 3-level hierarchical visualization that would grant users a broad overview of the full set of matchings while also allowing them to focus in on specific sections. The first layer would be the broad, one-screen overview; the second layer would show individual sections, and the third layer would show data related to one specific section including the original text. The raw text in particular we felt was a necessary feature since qualitative data on a section's topics, intent, or even keywords was unavailable.

A feature of the data we became attached to early on is how frequently sections of the PPACA, a Democratic bill that no Republican voted on, matched with sections of earlier Republican bills. Considering the public's current low opinion of congress we felt this evidence, however small, of bipartisanship was interesting and worth highlighting. The political party associated with a particular text became a theme throughout the whole design.

One Screen Overview

Our primary concern with the first layer of our visualization was to show an overview of all alignments in the 466 sections of the PPACA while still fitting it on the screen without scrolling. We went through several different approaches for what to show within each section, including an aggregated "Democrat" or "Republican" rating for each section based on the political association of its oldest matching text (figure 1). This was our first design, the idea of which came from the idea from the money chart visualization [2]. However, there were some disadvantages. First, for all the little squares on the left hand side, i.e. first layer, it was not clear to users that it was aggregated by party based on color; red for republican and blue for democratic. Secondly, the Grey boxes were originally meant to be sections that did not have any matches. However, there were also some sections associated with a third party that disrupted our three-color design. Thirdly, for the boxes with numbers on the right hand side, i.e. second layer, it was also not clear what the number exactly meant. In the figure the number is how many bills matched to that section. Many boxes had duplicate numbers that further confused people. Although each box in the second layer represented an individual section in PPACA, there was nothing that users could use to distinguish them.

Figures 2 and 3 were our intermediate designs. Rather than showing square boxes, we changed to a long column format where each row was a section. Figures 2 and 3 are both sorted by section ID. Figure 2 uses the same aggregation as figure 1. Figure 3 shows the ratio of blue democratic bill matches to red republican ones. Gray and white bars respectively show sections with no matches. We rejected design 2 since the aggregation was still not showing enough information to our liking. Design 3 was rejected since a sense of the absolute number of matches was missing.

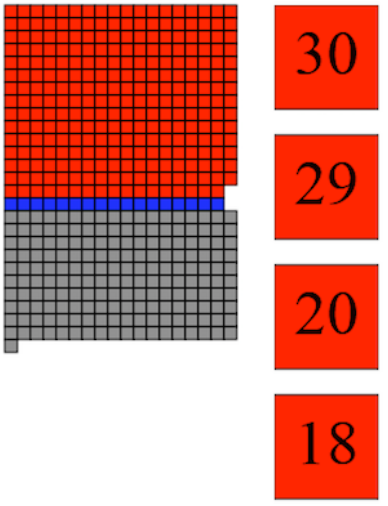

Figure 1

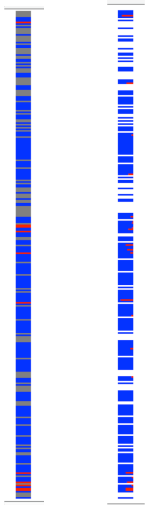

Figure 2 & 3

Alignment Chart

When focusing the visualization on a specific section, our initial plan was to show all the bills that matched to a given section as a bar graph. The vertical axis would be match strength and the horizontal would be time. At the suggestion of Nick Stramp though we switched to an alignment chart, which he had used in the past. The alignment chart seemed ideal for quickly showing where text was reused in the overall document, how often it was used, and when. The initial design included a linear time

scale. We implemented this and found the number of overlapping alignments to be a big problem. Too often multiple bills were introduced at the same time that all matched with the same section, sometimes across multiple sections. At the recommendation of Ham we decided to abandon the linear time scale, sort the alignments by time, and display them with even spacing. To still give a sense of time we binned matches in 3-month intervals.

The last property of the alignment chart we debated was the vertical axis. We found it hard to communicate what it was actually showing. We initially used the axis to show the percentage in the document where alignments started and ended. E.g. Alignments would start 25% into the text and end 75% of the way through the text. This proved to be confusing so we switched to simpler less detailed labels that just stated the top as the start of the text and the bottom as the end.

We also found some alignments to be such a small portion of the section length that they were difficult to see or click on. To solve this problem we had the chart 'zoom in' on the matchings by first showing their location in the full document and then automatically redefining the top and bottom axis to be the earliest match starting location and the latest match ending location respectively. This made all alignments large enough to be visible in all sections we could find.

Text Alignment

To best communicate how texts related to each other we decided to highlight the actual text that matched. The alignment data we had available included the matched string start and end locations but they turned out to be incorrect. Before the sections were matched all non-alphanumeric characters were stripped out for simplicity. That meant string locations no longer were correct in the original, formatted text. Our solution was to re-find the matching string in the original text. We replaced all non-alphanumeric characters in the original text with whitespace, rather than removing them entirely, to preserve the text length. We then placed "\s*" regex matches between each character in the match string to account for any number of spaces. The result was an inefficient algorithm that nevertheless allowed us to find the match string location in the original text.

Finding mismatch locations was a similar process. The original text was stripped of non-alphanumeric characters and the text was iterated through simultaneously with the two match strings. Characters in the original text not in the match string, mostly whitespace, were skipped and instances where the two match strings differed were marked as a mismatched character in the original text. We had access to gaps in the final matched string but adding them would have disrupted the formatting of the base text by changing the number of characters. Therefore we decided to not display gaps, though what is a gap in one text matching will appear as a mismatch in the other matching string so the difference is still made noticeable to a user.

RESULTS

One Screen Overview

Our current implementation consists of two bar charts. The left bar chart, first layer, shows the total sections of the PPACA whereas the right bar chart, second layer, shows the zoom in version of a particular region in the first layer. In the first layer, each row represented a section of the bill and divided into two regions, left blue is the number of matching bills for democratic and the right red is the number of matching bills for republican. Colors are taken from the New York Times' Over the Decades, How States Have Shifted visualization [1]. We felt it was a more honest representation than aggregation and displayed more information than a ratio. User can drag the window (white background) in the first layer to quickly navigate the sections region in the second layer. They can also click the grey area rather than drag the window. For the second layer, users can scroll up and down to find a particular section.

Sorting

Items in the first and second layers can be sorted not only matching numbers, but also by section ID and earliest match date. In order to support this feature, we add three buttons, which were SecID, Date and Match, on top of the first and second layer. The order of the matching sections in the first and second layer will rearrange based on the sorting.

Alignment Chart

Clicking on an individual section reveals an alignment chart showing all of the texts that had matches with that section. The chart's vertical axis shows where in the section text the matching started and ended, from the top down. The matchings are sorted chronologically by the matching bill's introduction date. Each alignment shows the matching bills' sponsor's political party with color and the bill's source, the house or senate, with fill. The chart also bins alignments in 3-month intervals and labels them as quarters to give users a rough sense of when bills were introduced. In instances where matching only occurs in a small subset of the original section text, the alignment chart will automatically zoom in on the relevant portion of the vertical axis where alignments occurred.

Text Alignment and Bill Information

By clicking on an individual alignment bar, a user can see the text itself. The section text for the ACA and the bill it matched to are both displayed. The full matched strings are highlighted yellow within the text. Areas where the two matching strings differed are highlighted in orange. We also added a feature for our collaborators to apply a text label to

any particular comparison, as the data is still being analyzed by them.

Clicking on an alignment also shows meta information about the matching bill including its name, when it was introduced, who sponsored it, the sponsor state and political party, whether the bill eventually became law, and a link to the complete text of the bill.

DISCUSSION

One Screen Overview

The purpose of the first and second layer was to organize and filter a large number of sections so that users could find one section to study in the third layer. As the methods section explains, we went through many different iterations of these sections and spent the majority of our design time focused here. The overall difficulty came from coming up with a design that showed a sufficient amount of detail for each section but also fit in a small portion of the screen.

Alignment Chart

The most drastic change the alignment chart underwent was the removal of the linear time scale. The decision was ultimately essential to avoid frequent alignment overlap. It took us little time to decide to zoom in on the alignment cluster and the rest of the design work was spent discussing how best to communicate what the alignment chart actually is. Our best solution was simple but careful labeling within the chart.

Text Alignment

Text size was another issue we dealt with. Considering the sheer amount of text we had to consider, the ACA alone is 900 pages, we had to rely on the formatting already present in the data. The best solution we found was to use the <pre> html tag to maintain the formatting while still allowing us to highlight text with 's. It meant the width of the text was dependent on the font size however. We made the text as large as we could without having to add horizontal scrolling. Ham suggested we cut the second layer in half to create more space for the text and add controls to adjust font size. We felt the juxtaposition of the first and second visual layers was too harsh with the second cut in half and horizontal scrolling in the text made it hard to navigate and read so we did not implement those suggestions but we did consider them.

CONCLUSION

Our goal was to design and build a visualization that allowed our collaborators to parse and understand the data they had collected efficiently. Based on feedback from them we have succeeded in this. We also hope that the visualization is intuitive enough that non-researchers can use and understand it. We ourselves entered this project with little understanding of the bill authoring process and came to know much more about it as we worked.

FUTURE WORK

This project was done as collaboration with other researchers and we plan to continue supporting the software with any bugs or small features they bring to our attention. We built the software to easily adapt to different data in the same format, and initial testing shows that it can, but if that turns out to be untrue we may make additional changes such as adding an administrator interface and refactoring the code.

ACKNOWLEDGMENTS

Many thanks to John Wilkerson and Nick Stramp for sharing their data with us, answering our numerous questions, and giving consistent, great feedback. Thanks also to Jeff Heer and Kanit Wongsuphasawat for teaching an excellent data visualization class.

REFERENCES

1. Bostock, Mike, Shan Carter. Over The Decades, How States Have Shifted. 2012. http://www.nytimes.com/interactive/2012/10/15/us/politics/swing-history.html
2. Munroe, Randall. Money Chart Visualization. 2011. http://xkcd.com/980/huge/#x=-5065&y=-6830&z=6
3. R. Kincaid, A. Ben-Dor, and Z. Yakhini. Exploratory visualization of array-based comparative genomic hybridization. Information Visualization, 4(3):176–190, 2005.
4. Wilkerson, J., Smith, D., and Stramp, N. Tracing the Flow of Policy Ideas in Legislatures: A Text Reuse Approach. *Unpublished.* 2014.

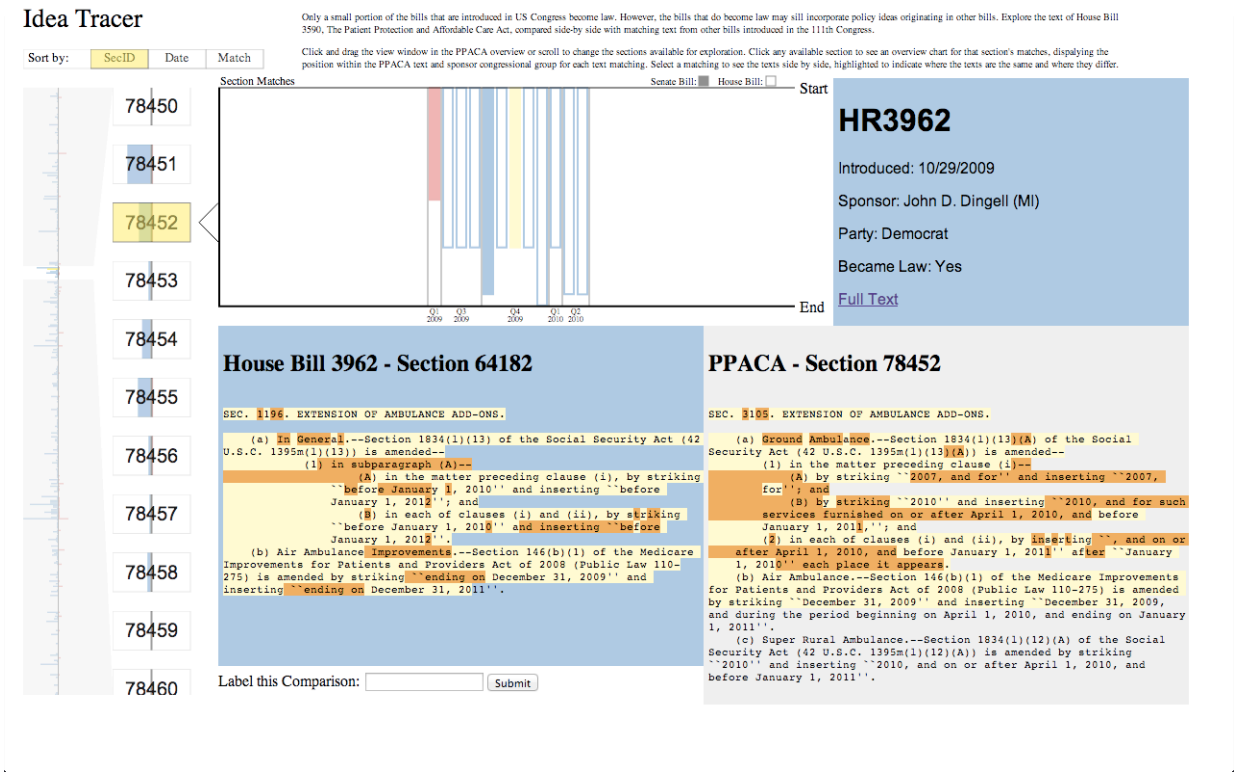

Figure 4: Screenshot of the final result