Selective Semantic Zoom of a Document Collection

Dustin Dunsmuir (dtd@sfu.ca)

Domain, Task and Data

Intelligence analysts often take on the challenging task of making sense of a large collection of documents. These documents are often unstructured text such as news articles or intelligence reports. This data is not as orderly as the typical database nor is it numerical so it is not easy to quickly summarize with a visualization. Analysts are interested in the activities of certain people or organizations mentioned within the documents so entity extraction systems have been developed that automatically extract the key people, places, dates, etc. from a large document collection (Calais [5] and MALLET [9] are two examples). Once this process is complete the result is a set of entities contained within each document.

Making sense of a document set involves building understanding of the relationships and the context of these relationships between different entities within the document collection. Often the task involves following up on a tip that mentions a specific person or event. In this case the analyst may start their analysis focused around a specific entity. At other times the analyst is exploring the data looking for anything suspicious and in this case they may wish to start with an overview. As an imitation of analyst's data, I will be using the VAST challenge data from 2006 called Alderwood within my semantic zoom document visualization. The documents are news articles.

Each document consists of paragraphs of text (usually 1-3 paragraphs). Tagged within each document is the set of entities (usually 1-3 word phrases). Each document also has an ID, and metadata: the date and sometimes the source of the document. This Alderwood dataset has had entities extracted, resulting in a single XML file with document elements containing elements for the ID, metadata, body text, and each entity. Each entity has an entity type (category such as "person") and a value. The same entity (same type-value pair) may occur in more than one document. We will make the assumption that each instance of these entities refer to the same real world entity.

Experience

I have been working in the area of visual analytics tools in an RAship for the last 8 months on the CzSaw system [8]. The main focus of CzSaw has been on capturing and supporting the analysis process through an underlying script of all actions, a history view generated from the script and a dependency graph that preserves the dependencies of the variables created in the analysis and allows quick propagation of changes. My focus on the project has been to develop the data visualizations including a hybrid view similar to a mix of the list view and graph view of Stasko's Jigsaw [13]. This project will form the base of my thesis work, and after the class ends, I will enhance it with more original techniques and perform user studies to compare it to well established work.

Previous Work

The Jigsaw system [13] is a visual analytics application designed to be used by intelligence analysts for sense-making across text documents. It provides multiple views of the data each designed to emphasize a specific aspect of it, but each within its own separate window. The CzSaw project [8] is being developed with a similar goal in mind and some additional functionality including the development of hybrid views designed to allow analyst the flexibility to use multiple representations of data within one view. The proposed information visualization technique will be a new hybrid view for the CzSaw system.

The data, consisting of documents and their contained entities, is a two level hierarchy. Node link diagrams are often the method of choice for displaying a hierarchy whether it be a tree or a general graph. One popular view within Jigsaw is the graph view used to see connections between documents and entities as edges between nodes in a node-link graph. Starting with a set of entities or documents, the graph is incrementally added to in layers by double clicking one of the entity nodes to see all of the documents it is contained in or double clicking a document to see all the connected entities. All edges exist only between a document and an entity so the fact that two entities are mentioned in the same document is seen by following a path of length 2. In the graph view of CzSaw we have allowed edges between any two entities or documents that have a relationship. The disadvantage with this technique is that all entities within a document will be connected with each other meaning that also possible edges between these entities will be present greatly increasing the total number of edges. The approach of this technique makes this relationship between entities very clear without any edges since the entities are both nested within a document (Figure 1).

Besides seeing the relationships between documents and entities it is also useful to be able to get an overview of the document collection. Jigsaw offers a document cluster view which shows each document as a small rectangle and clusters of documents can be seen matching a filter. Starlight [12] also offers the same representation where colour codings can be applied to the documents based on any values of the entities within the given document. IN-SPIRE [7] offers the galaxy view where each document is represented by only 4 pixels and situated in space dependent on the keywords within it. Clusters of documents are shown around common keywords. In any of these systems, to view any details about a document other than its name and the keywords of its clusters requires opening the document in another view. In this technique drill down will be possible directly within the view.

Zoomable User Interfaces (ZUIs) have been developed since 1993 to allow users to work with a large virtual space and navigate through it by zooming. The first such system was Pad [10] which has evolved through many iterations into the Piccolo2D toolkit [3]. Bederson also gives a large list of developed ZUIs in his latest paper [2]. These systems can be useful for investigating nested data structures by zooming into them. Some other research has investigated the use of focus+context with semantic zooming acting as a fisheye view. One of these techniques is the Continuous Zoom developed by Bartram et. al. [1]. A related method called variable zoom was used in a study done by Schaffer et. al. [11] involving subjects navigating a simulated telephone network. The fisheye view was compared to a full-zoom view and found to be faster to use and for some tasks allowed better performance. The ShriMP system was also developed for looking at nested graphs (software architecture) [14].

Proposed Solution

The new hybrid view, the selective semantic zoom view has two main functions:

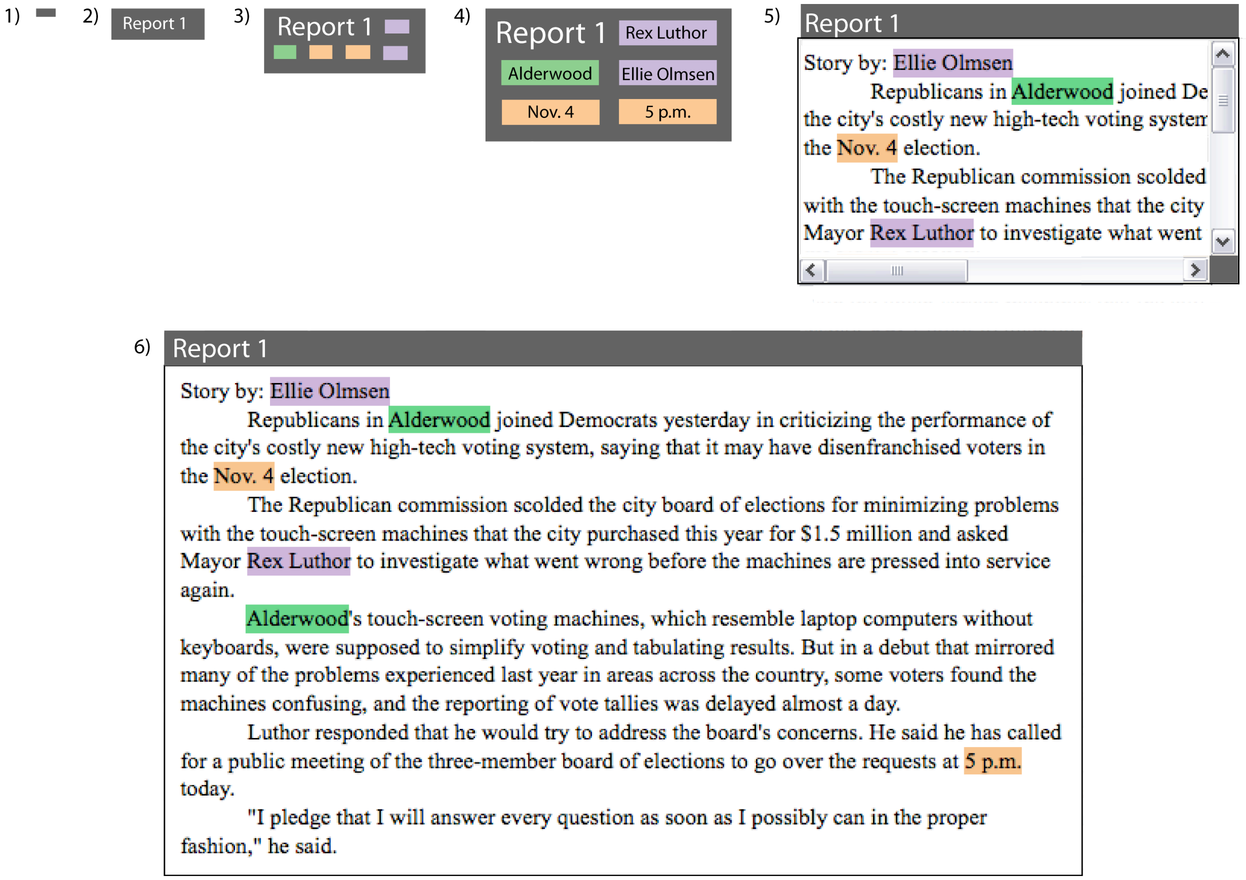
1) Show entities nested within documents so that the two level hierarchy of the dataset is intuitively shown. The relationship of "document contains entity" will be shown with actual containment. Brushing, positioning and colour can all be used to see all documents any given entity is contained within.

2) Allow an overview of the dataset and maintain context while allowing drill down and brushing across the view. Accomplish this through a selective semantic zoom. To do this, start by showing all documents at a very small size with no labels and then allow any subset of the documents to grow in size and change representation to show an ID (name), then the contained entities, and when fully expanded the full text of the document.

<u>High Priority Features</u>

One of the key interactions with the system will be to drill down into a document by semantically zooming it to see its contents. Rather than having to look at another view the analyst will be able to see different representations of documents all juxtaposed in one area. This allows the analyst to see any document as one of the following (Figure 1):

1) A small rectangle
2) A labelled rectangle
3) A labelled rectangle containing other rectangles (one for each entity, coloured by type)
4) A labelled rectangle containing labelled entity rectangles
5) A small window into the full document text which is scrollable to move around within the document.
6) The full text document. Within the document all the extracted entities are highlighted.



Story by: Ellie Olmsen
Republicans in Alderwood joined De
the city's costly new high-tech voting systen
the Nov. 4 election.
The Republican commission scolded
with the touch-screen machines that the city
Mayor Rex Luthor to investigate what went

Story by: Ellie Olmsen

Republicans in Alderwood joined Democrats yesterday in criticizing the performance of the city's costly new high-tech voting system, saying that it may have disenfranchised voters in the Nov. 4 election.

The Republican commission scolded the city board of elections for minimizing problems with the touch-screen machines that the city purchased this year for $1.5 million and asked Mayor Rex Luthor to investigate what went wrong before the machines are pressed into service again.

Alderwood's touch-screen voting machines, which resemble laptop computers without keyboards, were supposed to simplify voting and tabulating results. But in a debut that mirrored many of the problems experienced last year in areas across the country, some voters found the machines confusing, and the reporting of vote tallies was delayed almost a day.

Luthor responded that he would try to address the board's concerns. He said he has called for a public meeting of the three-member board of elections to go over the requests at 5 p.m. today.

"I pledge that I will answer every question as soon as I possibly can in the proper fashion," he said.

Figure 1 - The diferent zoom levels of a document.

Allowing this continuous semantic zoom of a document while keeping the context of documents around it at various zoom levels will be the main functionality developed in this application. What will be developed is an algorithm similar to the continuous zoom that moves documents to avoid occlusion when zooming is being done.

The second main feature that will be part of this application is the ability to select an entity in one document and see all the other documents it is in. This is necessary to allow investigation to focus on a specific entity to read all documents containing it and see the other entities in those documents. The basic method of showing these other documents will be to highlight each one by changing the border colour of the documents. This highlighting means that which documents contain the entity can be shown without opening them all.

The third main feature is the ability to perform a search for a given entity and have those documents that contain it be selected. This is necessary to allow the analyst to investigate a given entity they've received a tip on without first knowing of a document it is in (or where one of these documents is located in the view). Another necessary feature is the ability to filter out documents containing a given entity (through search or choosing directly) or filter out all the other documents in order to narrow down what is shown in the view. The analyst will also be able to choose an entity type and filter it out or all the other entity types from the view. This helps reduce the amount of data shown on the screen when the analyst wishes to have more space for a specific set of documents and/or entities. When highlighting is done after some documents have been removed, a message must be shown about how many hidden documents match the filter and the option given to add these documents back to the display.

Lower Priority Features

Below are a features that will be added to the visualization within the scope of the class project only if there is time. There are many possible feature additions to the technique, although it may be difficult to make them all work together without overloading the user's comprehension of the available interaction. Some more research will be done into the priority of the below features.

It would be useful to allow the emphasis of a subset of documents to be stronger than the mentioned highlighting. Different possibilities include:

- Changing spatial position to cluster all documents with a given entity together. With the different sizes of documents this will involve some non-trivial layout algorithm.
- Changing the background colour of the document. This is a very strong change and may conflict with the individual entity colours shown within.
- Allow a set of documents to move back and forth briefly. This technique will instantly make the documents stand out from the rest, but should not be used for long periods of time.
- A more advanced feature is that a given set of documents can be merged into one nested region that represents them all. Then some entities will be contained multiple times so these repeats will be removed, instead varying the size of each repeated entity and a treemap could be used to clearly show the relative number of repeats. In this way entities sharing many documents could be found. Note though that without this feature, repeated entities may aid the analyst by making it easy to see that multiple documents containing the same few entities.

Each of the above modalities for separating groups of entities from the rest could be used for a different type of entity. For example spatial position could be used to group together documents containing the same *place* entities. The only trouble comes when two places are mentioned in the same document.

Since the one view is designed to show different representations and support both overview and focus it can be an area for the analyst to organize their findings. In this way it would be good to allow an analyst to move and organize documents into different areas of the screen to support their spatial memory. To do this the layout will have to be much more adaptive.

In the design so far there is no intuitive automatic starting position of the documents. Given enough time it would be useful to position documents based on shared keywords automatically using text analytics techniques as in IN-SPIRE [7].

In the design so far there is also no intuitive location for each entity within the document. This could be in a location relative to where it is highlighted in the full text of the document (since when zoomed in further this is what appears). The challenge here is that many entities may be in a small part of the document while other parts of the document have few and an entity may be mentioned in the document more than once.

Scenario

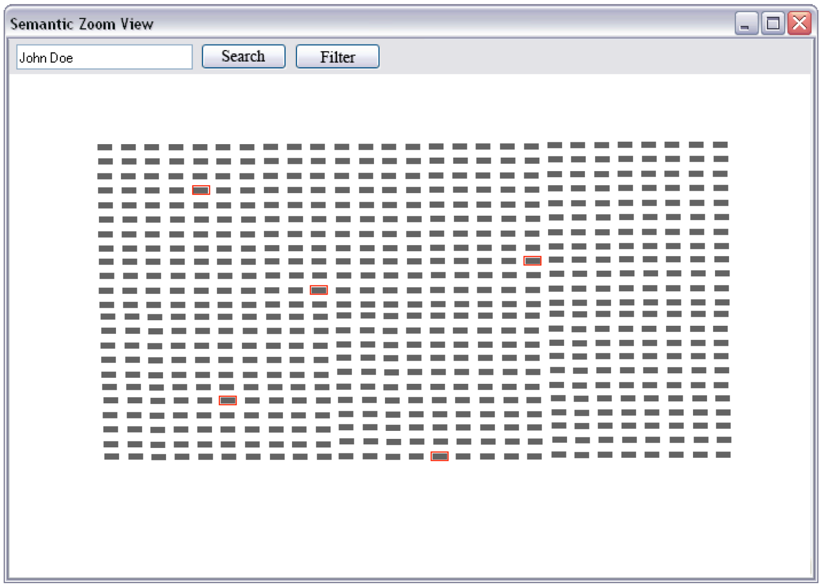


Figure 2 - Performing a search in the view.

The analyst receives a tip about a suspicious person named John Doe. They have a document set loaded within the Semantic Zoom view so that each document is shown in the view as a tiny rectangle.

The analyst searches for John Doe using the search bar at the top of the window. This causes all the documents containing John Doe to be selected (Figure 2). There are 5 documents. The analyst then uses the scroll wheel of the mouse to expand (zoom up, drill down into) all selected documents (the 5 with Doe). The analyst zooms just to the point where he can see all the entities labelled within each of the documents.

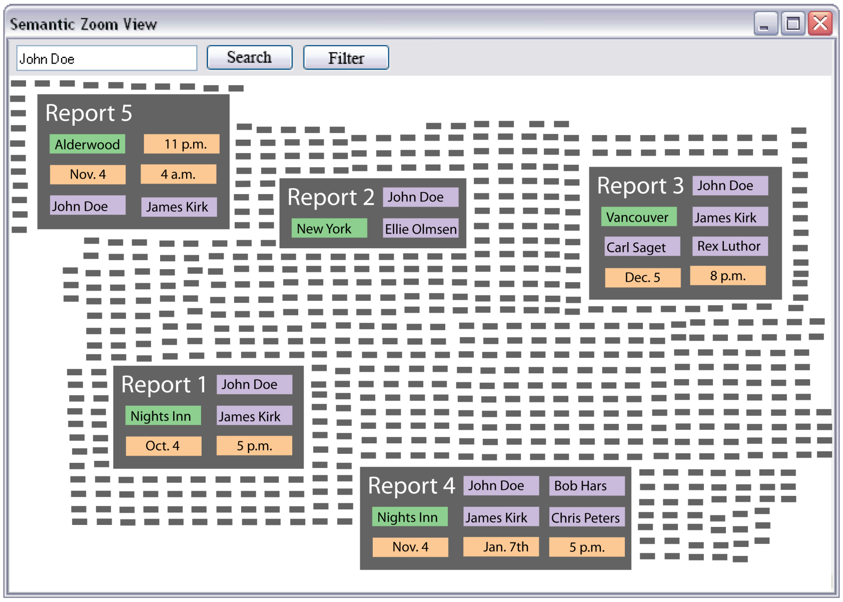


Figure 3 - Expanded Documents. Note that this is not exactly how the layout will move other documents but this algorithm has not been worked on yet so this is just a rough mockup of it.

He notices that James Kirk is mentioned in 4 of these documents. He clicks Kirk to see all the documents containing him highlight. These include only the 4 zoomed documents and 2 others. Thus he concludes that Kirk and Doe are fairly strongly tied together but to confirms this and get the context he must read the documents.

To give more space in the view he zooms all 5 documents down to their label (so they still stand out). Then he in turn zooms each of the 4 including Kirk to full document size, reads the document and then zooms it back to just the label. From the documents he learns that the two men used to work together and since then have been in attendance at many of the same events.

With the 4th document though he notices the name of a hotel, Nights Inn, that looks familiar. Instead of zooming out this document, he clicks on Nights Inn within the document text and sure enough, two of the documents he has already read are highlighted along with a few other documents.

He can now explore in a similar manner these new documents looking into other entities mentioned in many of them, reading the documents, and building an understanding of activities that John Doe may be involved in. When he wishes to narrow down the investigation he can filter out documents by selecting them, right clicking and choosing filter from the popup menu.

Implementation

This information visualization project will be a new view within CzSaw that will use the database and data query methods of CzSaw but other than that will be a completely separate set of code for the purpose of this class. No one else on the CzSaw team will have access to this set of code or work on it and some updates to the data querying class may be done by me to enable functions of this project.

To implement this technique I will use the ZVTM (Zoomable Visual Transformation Machine) toolkit that is centered around zooming. It is a Java library and the rest of CzSaw is also written in Java. I have already experimented with the ZVTM library and created the basic zoom levels of a single document. To be able to zoom documents independent of the rest of the view each document will have to be on it's own virtual space and then portals to these spaces are within the view.

Expanding or contracting documents will be a central component of the system and so we will need a continuous zoom layout algorithm which works quickly with a large number of items but also is intuitive to the user. The algorithm may not be as complicated as Shrimp [14] or the Continuous Zoom [1] since there are no edges involved but the implemented algorithm will still need to be intuitive to the user while not quickly taking up unneeded amounts of space.

Colours used in this visualization will be taken from the ColorBrewer [4] qualitative palettes. It is designed for cartography but this technique also involves many areas (rectangles) of colour close to each other. The colour scheme used will also be checked on the VisCheck website [6].

Milestones

November 6th: Have the documents being displayed in the view with nested entities and be able to display them at different sizes with semantic zoom but without any decent algorithm responding to the change in size. That is occlusion will not yet have been removed.

November 18th: (For project update presentation). Finish basic layout of documents and entities within them, so that no occlusion occurs..

November 27th: Have filtering and brushing working in the system.

December 11th: Finalize the layout algorithm and add any lower priority features that there are time for.

References

[1] Bartram, L., Ho, A., Dill, J., and Henigman, F. (1995) The continuous zoom: a constrained fisheye technique for viewing and navigating large information spaces. *Symposium on User Interface Software and Technology.* pp. 207-215.

[2] Bederson, B.B. (2009) The Promise of Zoomable User Interfaces. *CHI 2009,* April 4–9, 2009, Boston, Massachusetts, USA.

[3] Bederson, B.B., Grosjean, J., and Meyer, J. (2004). Toolkit Design for Interactive Structured Graphics. *IEEE Transactions on Software Engineering, 30 (8),* pp. 535-546.

[4] Brewer, C.A., Hatchard, G.W., and Harrower, M.A. (2003) ColorBrewer in Print: A Catalog of Color Schemes for Maps. *Cartography and Geographic Information Science, 30*.

[5] Calais. (2009) Developed by Thomson Reuters. http://www.opencalais.com/

[6] Dougherty, R., Wade, A. (2002) VisCheck. http:www.vischeck.com.

[7] IN-SPIRE. (2008) Developed by the Pacific Northwest National Laboratory. http://in-spire.pnl.gov/getacopy.stm

[8] Kadivar, N, Chen, V., Dunsmuir, D., Lee, E., Qian, C., Dill, J., Shaw, C., and Woodbury, R. (2009) Capturing and Supporting the Analysis Process. Proceedings of IEEE Visual Analytics Science & Technology 2009, Atlantic City, NJ, Oct 11-16, 2009, pp. 131-138.

[9] McCallum, A. K. (2002) MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.

[10] Perlin, K., and Fox, D. (1993) Pad: an Alternative Approach to the Computer Interface. *International Conference on Computer Graphics and Interactive Techniques* pp. 57-64.

[11] Schaffer, D., Zuo, Z., Greenberg, S., Bartram, L., Dill, J., Dubs, S., and Roseman, M. (1996) Navigating Hierarchically Clustered Networks through Fisheye and Full-Zoom Methods. *ACM Transactions on Computer-Human Interaction 3 (2).* pp. 162-188.

[12] Starlight (2008) Developed by the Pacific Northwest National Laboratory. http://starlight.pnl.gov/

[13] Stasko, J., Gorg, C., and Liu, Z. (2008) Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization 7,* pp. 118-132.

[14] Storey, M-A., Best, C., and Michaud, J. (2001) SHriMP Views: An Interactive Environment for Exploring Java Programs. *International Conference on Program Comprehension.*