Visualising text co-occurrence networks

HIRSCH, Laurie and ANDREWS, Simon

Visualising Text Co-occurrence Networks

Laurie Hirsch

Simon Andrews

Sheffield Hallam University

Abstract. We present a tool for automatically generating a visual summary of unstructured text data retrieved from documents, web sites or social media feeds. Unlike tools such as word clouds, we are able to visualise structures and topic relationships occurring in a document. These relationships are determined by a unique approach to co-occurrence analysis. The algorithm applies a decaying function to the distance between word pairs found in the original text such that words regularly occurring close to each other score highly, but even words occurring some distance apart will make a small contribution to the overall co-occurrence score. This is in contrast to other algorithms which simply count adjacent words or use a sliding window of fixed size. We show, with examples, how the network generated can be presented in tree or graph format. The tree format allows for the user to interact with the visualisation and expand or contract the data to a preferred level of detail. The tool is available as a web application and can be viewed using any modern web browser.

1 Background

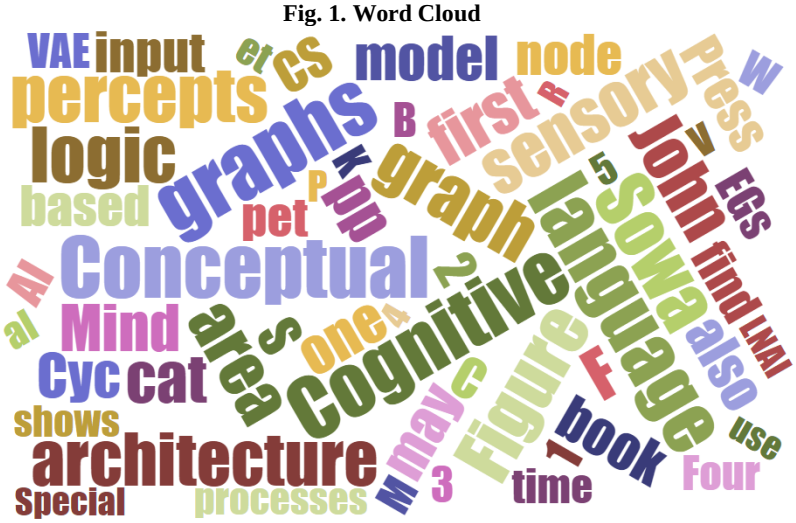
Visual representations have proved to be useful alternatives to linear text documents. The mind mapping technique was introduced in the 1960s and is thought to encourage learning. However, creating mind maps can be a complex and time-consuming undertaking and the ability to automatically produce text visualisations has attracted significant research in recent decades. A number of possible benefits have been attributed to such tools including managing information overload, providing summaries and 'impression formation'. Tools have been developed for identifying topics and topic correlations, displaying knowledge and generating concept clouds [1][2]. Here we will briefly outline a number of existing techniques and then show how we have developed a method based on word co-occurrence which can be used for generating both graphs and trees in various types of diagram. Here we include a number of example visualisations, all of which are based on the text of a paper concerning conceptual structures[3][1]

[1] Available at http://www.jfsowa.com/pubs/ca4cs.pdf It may help the reader to briefly read the article before viewing the visualisations.

1.1 Word Clouds

Although many systems are formed using user provided tags, there has been significant interest in ‘word tags’ or ‘text tags’ which are automatically generated using the text found in documents or web sites. The popular tool Wordle [4] has seen a steady increase in usage and many variations have been made available. Word clouds are based on the frequency of individual words found in the available text after stop word removal. The most frequent words are selected and then presented using various techniques to adjust font, colour, size and position, in a way that is pleasing and useful to the user. The words are often sorted alphabetically, although various systems of arrangement have been proposed and attempts have been made to place similar words together. Word clouds are simple and are commonly presented on web sites with little or no explanation of how they should be used or interpreted. A word cloud of the Sowa text can be seen in Figure 1[2]

Fig. 1. Word Cloud



A commonly cited issue with word clouds is that they can hinder understanding due to the fact that they lack information about the relationships between words.

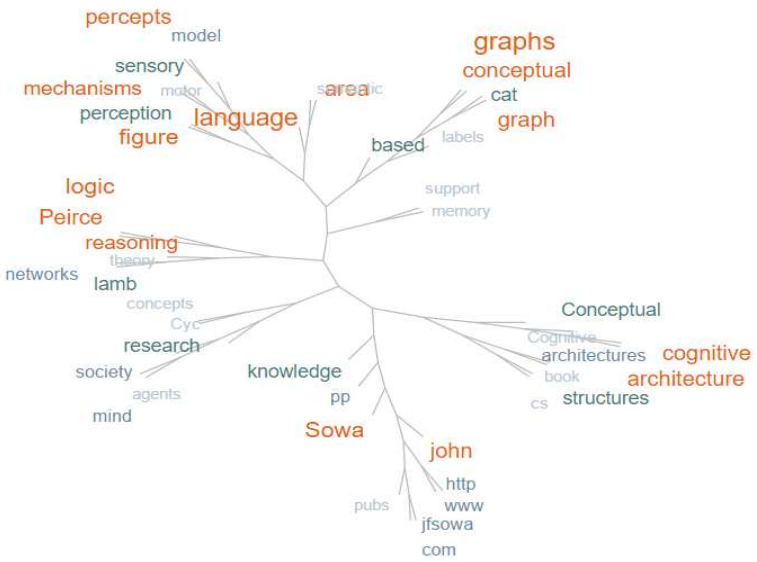
1.2 Tree Clouds

Trees have been presented as an easy to read and meaningful format and the term 'tree cloud' has been proposed [5]. A freely available system which generates trees based on the semantic distance between words derived from the original text is also

[2] Created using the tool at https://www.jasondavies.com/wordcloud/

available and gives the user an indication of the relationship between the key terms in the visualisation. The Sowa text produces the tree cloud shown in figure 2[3]

Fig. 2. Tree Cloud



The tree cloud includes colouring, font sizes and arcs to indicate relationships between topics.

2 Description of the System

In this section we will describe how our system known as txt2vz (http://txt2vz.appspot.com/) works and will compare visualisations produced with other text visualisation tools.

2.1 Pre-processing

To reduce dimensionality of the document(s) all words are placed in lower case, stop words are removed and stemming applied, such that only the most frequent form of a word is preserved.

2.2 Significance Measure.

We define a measure of significance for a pair (P, Q) of words, based on the number of occurrences of (P, Q), or more specifically the co-occurrences and the distance between

[3] Using the tool at http://treecloud.univ-mlv.fr/cgi-bin/NuageArbore_EN.cgi#

P and Q where the distance between P and Q is defined to be the number of words between P and Q:

$$significance(P,Q) = \sum_{i=1}^{M} B^{distance(PQ_i)} \quad (1)$$

where M is the number of co-occurrences of P and Q; is the distance between P and Q in the ith co-occurrence; 0<B<1 B is between 0 and 1 and typically set to 0.9. We do not consider the significance if the distance is beyond a pre-set maximum distance which has a default of 20 words. The use of a decaying function here is in contrast to commonly used 'sliding window' methods of computing co-occurrence where we simply count the number of times that two words occur within a predefined distance.

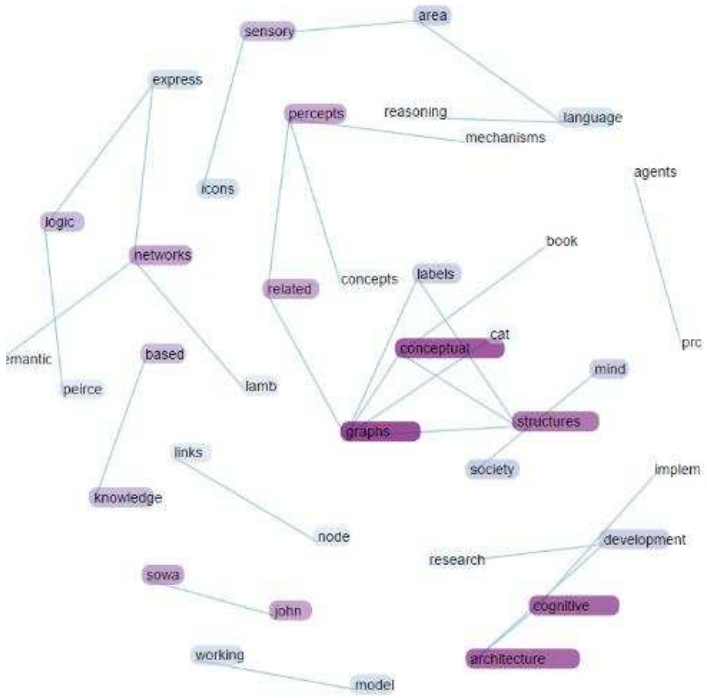
2.3 Network Generation Algorithm.

The visualisations produced by txt2vz are intended for use in a web application and the visualisation should be presented to the user in a reasonable time. Even after dimension reduction there are likely to be a large number of unique words in a text and computing the co-occurrence value for each possible pair can be time consuming. We therefore sort the words according to frequency and select the top N words for the next stage where N is typically set to 200. The significance of each pair of words from the remaining set is computed and all the word pairs are sorted in descending order by their accumulated co-occurrence values. An undirected graph can then be built by selecting the top K word pairs and creating an edge between the two words of each pair. The graph is built using the d3.js software library [6]

2.4 Txt2vz network

The simplest format for txt2vz has been described previously [7] and the visualization for the Sowa text is shown in Figure 3.

Fig. 3. Txt2vz graph



The visualisation is made up of a number of undirected graphs. Links between words are shown and the most highly linked words should move toward the centre of the graph.

The three diagrams shown above include many common words but have a number of important differences. For example, the fact that 'conceptual' and 'graphs' are related is not clear from the word cloud in Figure 1. We have argued that links between words in the txt2vz graph are made clearer via the arcs in the graph when compared to the tree cloud in figure 2. The graph shown above is actually animated on the web page using d3's force directed layout.

2.5 Txt2vz Tree

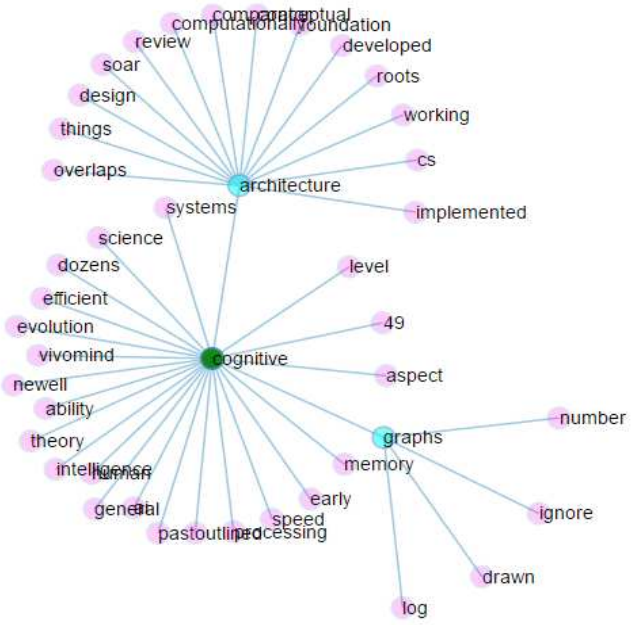
We have been experimenting with new visualisations based on the same co-occurrence data but in tree format. A graph such as the one shown in figure 3 may contain many possible trees. In our system the list of word pairs used to generate the visualisation is already in order of the co-occurrence value for the pair. A relatively simple way of creating a tree is to identify the root as the most frequently occurring word of the first pair. Iterating through the list we build up the tree, pair by pair, but only create new nodes if a tree structure is preserved. If this is not the case the pair is discarded. We

argue that there are certain advantages of txt2vz trees over those produce by tree cloud (Figure 2). Firstly, the visualization makes clear the links between two nodes, and secondly where the tree branches the arcs emanate from a labeled node making the relationship obvious.

Force directed collapsible tree.

Another advantage of the tree format is that it is relatively easy to make the visualisation collapsible so that the user can interact and expand or collapse non-leaf nodes as required to obtain a useful level of detail. A fully expanded version of the visualized Sowa text is shown in figure 4 but if the user clicks on an internal node the tree will collapse into the node until the user clicks again.

Fig. 4 Expanded tree



.

Radial Reingold–Tilford Tree.

A possible disadvantage of the force directed tree shown in Figure 4 is that the text for the nodes can overlap, particularly for lager trees. Although the user can interact with the diagram and pull the nodes apart, it is not always an ideal way to get a quick overall impression of the structure. An alternate radial tree format based on the Reingold–Tilford algorithm [8] is also available, and an example of the Sowa text is shown in figure 5 and 6. In this case the nodes and labels are spaced in rather a beautiful way.

Fig. 5. Radial Tree (small)

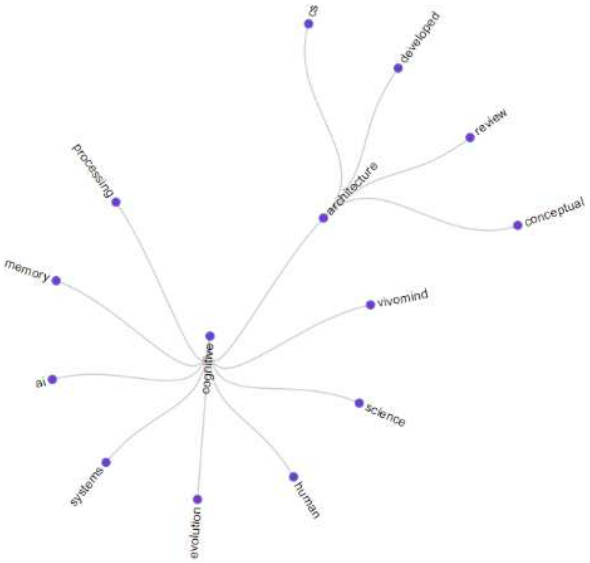
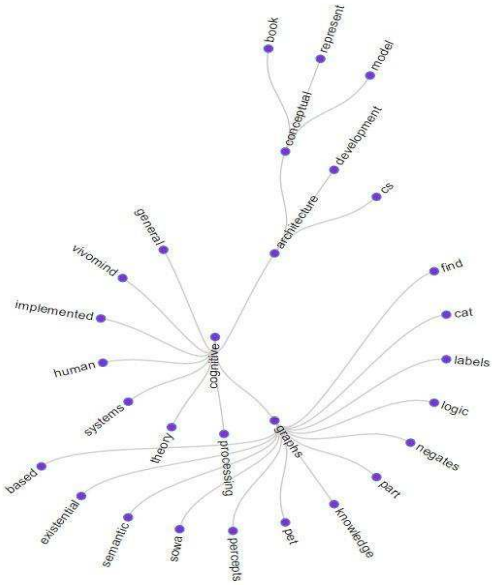


Fig. 6: Radial Tree



3 Discussion and further work

Txt2vz is work in progress. Apart from the basic graph types there are many parameters that affect the appearance of the final visualisation. We have therefore added a control panel to the web application so that the user can experiment and view different perspectives on the same piece of text. For example, reducing the number of word pairs to be analyzed will produce the smaller graph shown in figure 5, which some users may prefer. We would not want to argue that one particular graph type or cloud is the 'best' but we do suggest that it may be useful for the user to be able to be able to switch between different types of visualization.

We would like to spend more time evaluating the usefulness of the tool as perceived by human subjects. We also hope to investigate the feasibility of using Txt2vz as part of a web search engine such that a user could be presented with a quick visual summary of the content of the pages pointed to by the result links. Lastly we are investigating the possibility of scaling txt2vz such that it can produce visualisations of large text datasets. We leave the reader with two further radial tree visualisations on different texts. Figure 7 shows a visualization of this article and Figure 8 shows a visualization of Darwins 'On the Origin of Species'.

Fig. 7

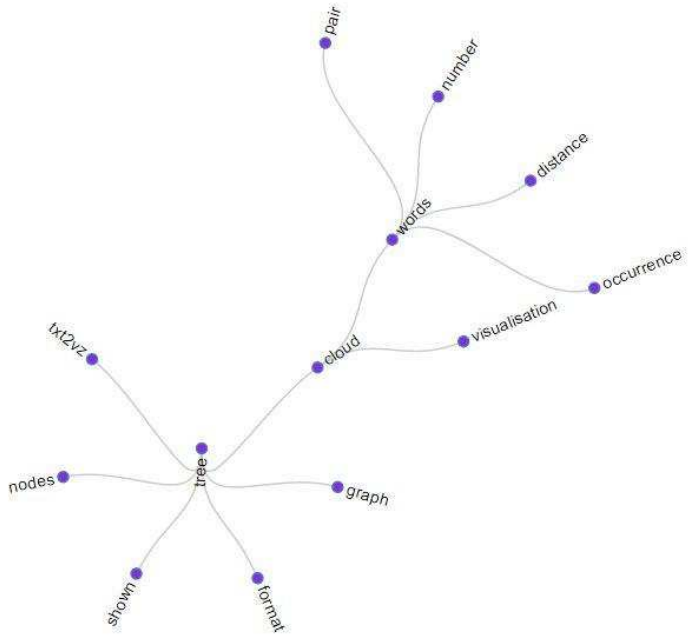
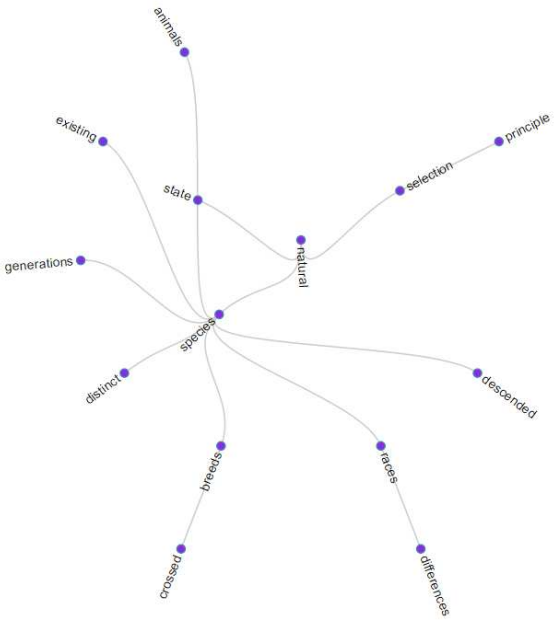


Fig. 8. The Origin of Species



4 References

1. Smith, A., Chuang, J., Hu, Y., Boyd-Graber, J. and Findlater, L., 2014. Concurrent Visualization of Relationships between Words and Topics in Topic Models. Sponsor: Idibon, 79.
2. Aga, R.T. and Wartena, C., 2015, October. Constructing concept clouds from company websites. In Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business (p. 38). ACM
3. Sowa, J.F., 2011. Cognitive architectures for conceptual structures. In Conceptual Structures for Discovering Knowledge (pp. 35-49). Springer Berlin Heidelberg.
4. Viégas, F.B., Wattenberg, M., Tag Clouds and the Case for Vernacular Visualization, ACM Interactions, XV.4 - July/August, 2008
5. Gambette, P. and Véronis, J., 2010. Visualising a text with a tree cloud. In Classification as a Tool for Research (pp. 561-569). Springer Berlin Heidelberg..
6. Bostock, M., 2014. Data-Driven Documents-D3. js.
7. Hirsch, L. and Tian, D., 2013, January. Txt2vz: a new tool for generating graph clouds. In International Conference on Conceptual Structures (pp. 322-331). Springer Berlin Heidelberg.
8. Reingold, E.M. and Tilford, J.S., 1981. Tidier drawings of trees. Software Engineering, IEEE Transactions on, (2), pp.223-228.