Txt2vz: a new tool for generating graph clouds

HIRSCH, L <http://orcid.org/0000-0002-3589-9816> and TIAN, D

Txt2vz: a New Tool for Generating Graph Clouds

Laurie Hirsch

David Tian

Sheffield Hallam University, Sheffield, UK
l.hirsch@shu.ac.uk

Abstract. We present txt2vz (txt2vz.appspot.com), a new tool for automatically generating a visual summary of unstructured text data found in documents or web sites. The main purpose of the tool is to give the user information about the text so that they can quickly get a good idea about the topics covered. Txt2vz is able to identify important concepts from unstructured text data and to reveal relationships between those concepts. We discuss other approaches to generating diagrams from text and highlight the differences between tag clouds, word clouds, tree clouds and graph clouds.

Keywords: visualization, concept map, tag cloud, tree cloud

1 Introduction

Tag clouds are simple visualizations that display word frequency information via font size and colour, that have been in use on the web since 1997. Users have found the visualizations useful in providing an overview of the context of text documents and web sites. Whereas 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. For example, the popular tool Wordle has seen a steady increase in usage [1]. 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 commonly sorted alphabetically, although various systems of sorting and arrangement have been proposed and attempts have been made to place similar words together [2] [3]. Word clouds are simple and are commonly presented on web sites with little or no explanation of how they should be used or interpreted. Three distinct tasks have been identified which may be accomplished namely, searching, browsing and "impression formation" whereby "The cloud can be scanned to get a general idea about a subject" [4]. Successful realisation of this last task is the main objective of the Txt2vz tool. Trees have been presented as an easy to read and meaningful format and the term 'tree cloud' has been proposed. A freely available system which generates trees based on the semantic distance between words derived from the original text is also available [5].

Co-occurrence information has long been understood to be an important aid to understanding the meaning of words, and using this information has proved essential to many natural language processing and information retrieval tasks [6] [7]. We extract and use co-occurrence information here as a way of giving context to words presented to the user and as a way of identifying and highlighting the most important words. We propose a new method of generating diagrams, based on co-occurrence information derived from the original text. We suggest the term 'word graphs' for the Txt2vz generated diagrams since they are not necessarily in tree format and indeed can sometimes be in the form of two or more disjoint graphs. An important feature of the Txt2vz graphs is that link information is the critical element of graph construction: co-occurrence links are directly displayed and nodes (words) with the most links are placed toward the centre of the graph.

2 Description of Txt2vz

2.1 The Overall Methodology.

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. Depending on the size of the document or the collection, this can still leave a large number of words, and further reduction is achieved by ordering words according to their frequency or tf-idf (term frequency-inverse document frequency) weighting in the case where the document is part of a collection, and then selecting the top N words from the sorted list.

After dimension reduction, every possible pair of the remaining words is analysed for co-occurrence information. Many techniques have been described for identifying co-occurrence [6] [7] but we take a relatively simple approach here. A graph is generated by selecting the top K pairs of words from a list of word pairs in descending order of their significance value defined as follows.

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 *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*; distance(PQ_i) is the distance between *P* and *Q* in the *i*th co-occurrence; 0<*B*<1 *B* is 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.

2.3 Graph Generation Algorithm.

The significance of each pair of words is computed and all the word pairs are sorted in descending order by their significance values. An undirected graph is then built by selecting the top K word pairs in the rank and creating an edge between the two words of each pair. The degree of each node (word) i.e. the number of edges attached to each node, can be used as an indication of the importance of that word. The most significant word is the word with the largest degree. Different colours are used to group words of similar importance and node and font sizes are used to highlight the importance of the words within the graph.

The type of graph produced can be partly determined by the user. In particular we have provided an adjustment facility whereby the user can change the number of words (N) to analyse; the number of links to display (K) and the maximum distance allowed between words when calculating co-occurrence (shown in figure 1).

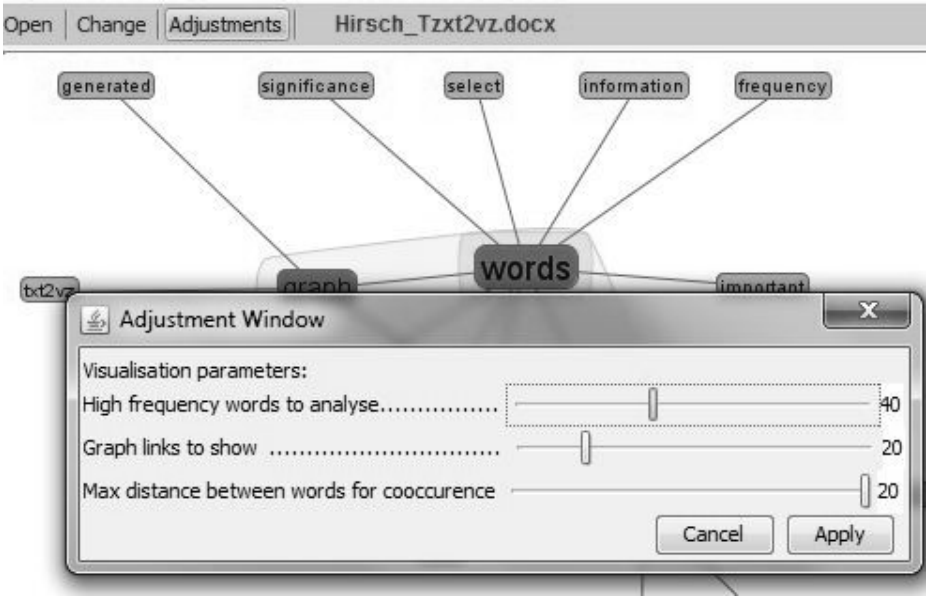

Fig. 1. Txtvz adjustment window

Algorithm

1. Tokenize the text and apply dimension reduction using lower case, stop words and word stemming.
2. Order the words according to frequency or tf-idf where the document is part of a collection.
3. Create a set of words W by selecting the top N words from the ordered list where N has a default value of 40 but can be adjusted by the user.

4. Analyse every possible pair of words from W and assign a co-occurrence value to each pair. For each case where both words occur within a maximum distance of 20 words (this value can also be adjusted by the user) we add a value to the co-occurrence metric for the pair determined by:

$$0.9^{wordDistance}$$

where *wordDistance* is simply the number of intervening words. Note: a decaying function is used such that words occurring closer to each other add more to the co-occurrence value.
5. Create an ordered list of the word pairs based on the co-occurrence value for each pair.
6. Generate a graph by selecting the top K word pairs from the sorted list of pairs where K is a value that can be set by the user, but with a default value of 20.
7. The number of links attached to each node is used as a further indication of the importance of a particular word.

As an initial example if we use the text taken from the ICCS'13 call for papers (http://iccs2013.hbcse.tifr.res.in/call-for-papers) and show the top 10 pairs

Table 1. ICCS'13 CFP co-occurrence values

Word Pair		**Co-occurrence value**
data	stem	8.277417941925659
conceptual	structure	7.837253642946149
papers	conference	6.667569657552907
papers	accepted	6.645902314469966
papers	called	6.5704478047496115
papers	phd	5.7306734434201365
conceptual	knowledge	4.425574377763965
data	concept	4.275419763354885
called	workshops	4.2135456501
dot	chair	4.205350186716726

The word pairs are used directly to create the graph shown in figure 4. Each unique word generates a node and each pair generates an edge.

3 Examples.

We begin by presenting diagrams generated from the ICCS'13 call for papers which contains 493 words. We compare the graph produced by Txt2vz with the ones generated by the Wordle [8] and tree cloud [5] approaches.

Fig. 2. Wordle word cloud of ICCS'13 CfP

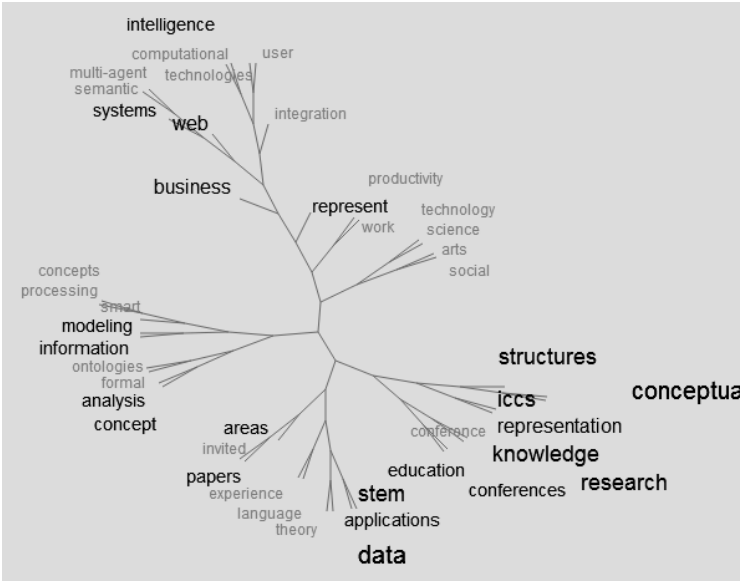

Fig. 3. Tree cloud diagram for ICCS'13 CfP

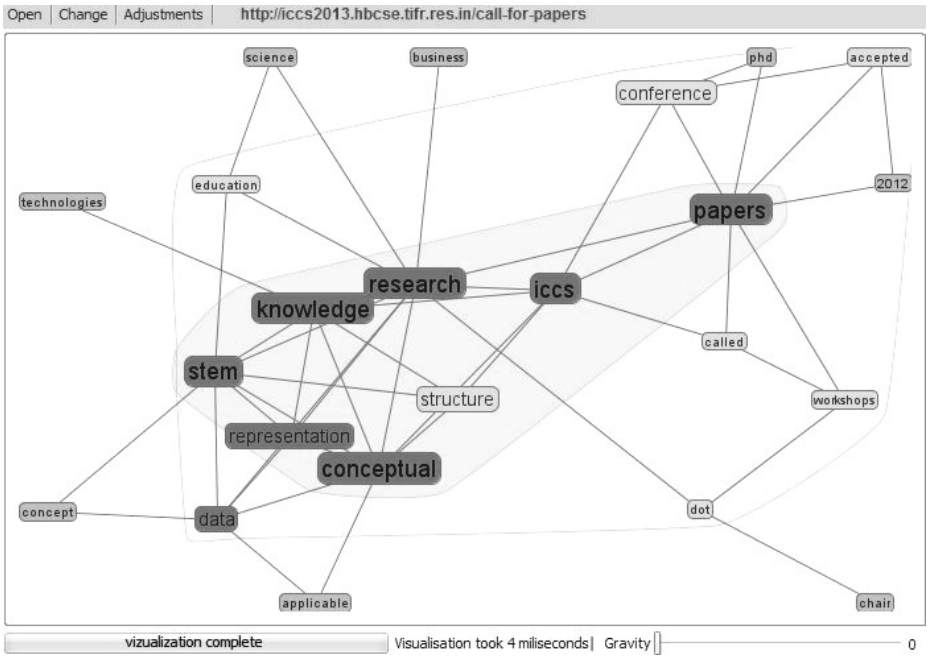

Fig. 4. Txt2vz graph cloud for ICCS'13 CfP

The three diagrams shown in figures 2, 3 and 4 include many common words, but the presentation is different in a number of respects. Which format is the 'best' is not a discussion we plan to resolve in this paper. However, we can identify that certain types of information can be obtained from the different formats. For example the fact that 'conceptual', 'knowledge' and 'structures' are related is not shown on the Wordle diagram whereas it is evident from the positioning of the words in the tree cloud and made clear via the arcs in the Txt2vz graph. Links between words in a Txt2vz graph indicate recognizable connections and nodes with a higher number of links are emphasized using large font sizes and by positioning these nodes at the centre of the graph. In figure 4 the word 'chair' has only one link and appears smaller and to the edge of the window whereas 'iccs' has 7 links, is larger and placed toward the centre of the graph. The point we wish to emphasise here is that the Txt2vz diagram clearly shows how words link to each other and uses that information to highlight important topic words with a high number of links, rather than only using word frequency information as in the tree cloud. For example, the Txt2vz diagram makes it easy to see that 'conceptual' is directly related to a number of other words and this is not obvious from the other two diagramming systems. You can test your own documents at txt2vz.appspot.com

4 Large Documents and Adjustments

Txt2vz uses Apache Lucene for indexing documents and for calculating the co-occurrence values and the Prefuse (http://prefuse.org/) library is used for graph generation. Lucene scales very well and large documents and collections can easily be visualized in a short time frame. Graphs can be significantly varied via user adjustments. Figure 5 and 6 show visualizations for Darwin's 'Origins of Species' using 50 links and 1 link respectively.

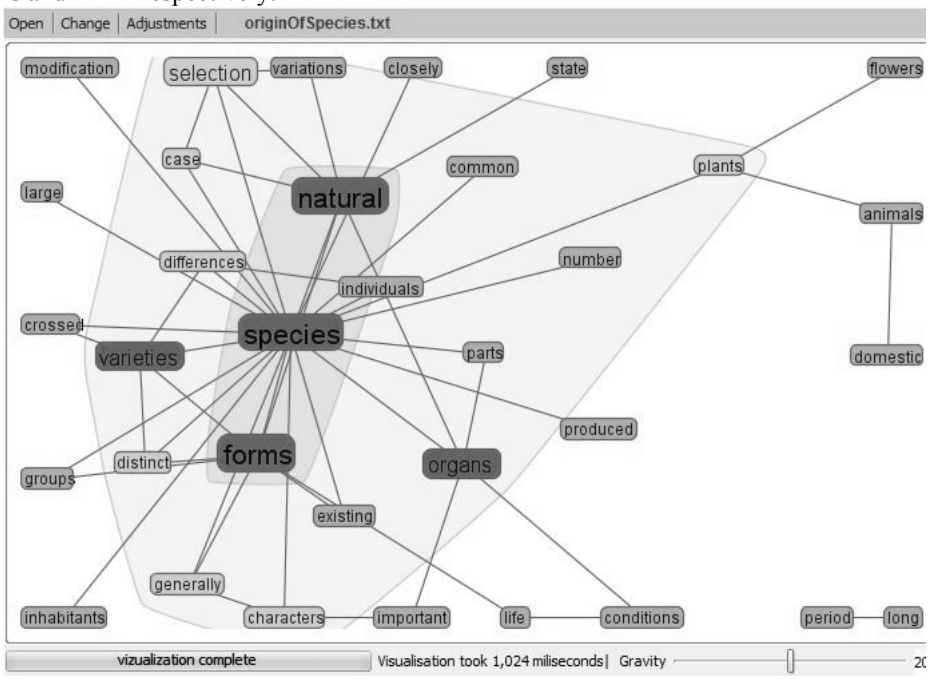

Fig. 5. Origins with 50 links

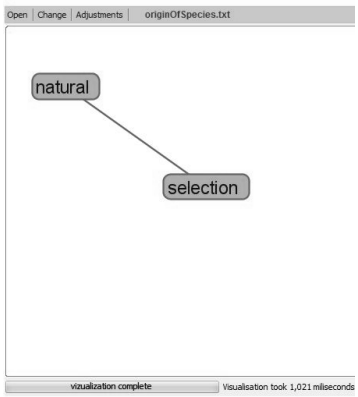

Fig. 6. Origins with 1 link

Txt2vz also offers an alternate radial graph format which uses the Docuburst library[9] and we show the visualization of this paper in radial graph format (figure 7).

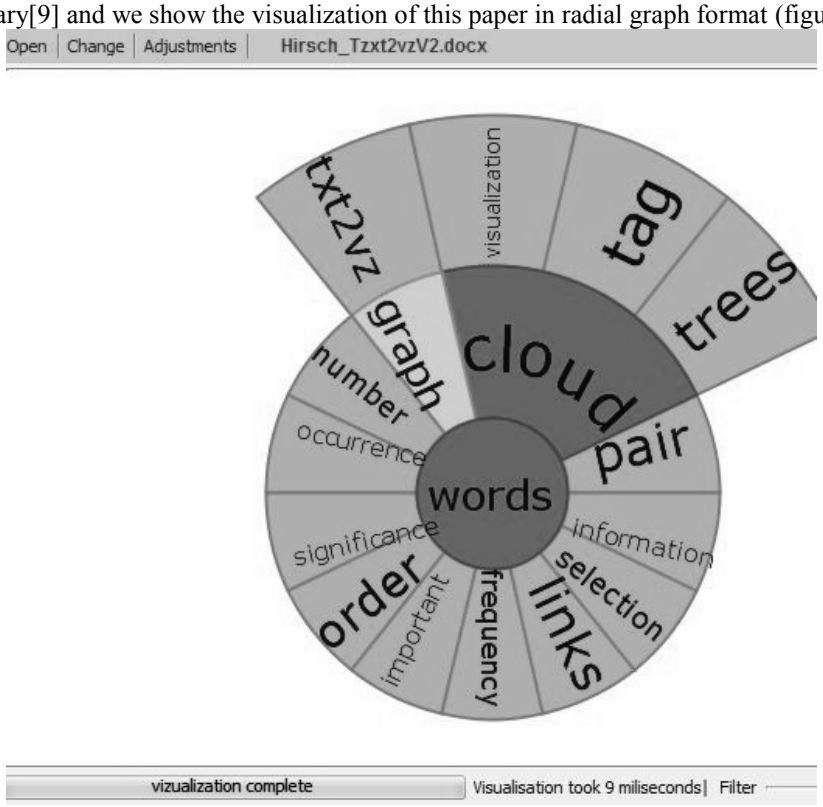

Fig. 7. This paper in radial graph format

5 Document Collections

We believe that the graphs produced by Txt2vz might be especially useful to people who need a visual summary of large collections of documents and as mentioned above, Lucene makes this perfectly possible. The example shown in figure 8 was generated in less than 10 seconds from 389 documents from the training set for the Reuters-21578 [10] 'crude' category containing news stories concerning crude oil.

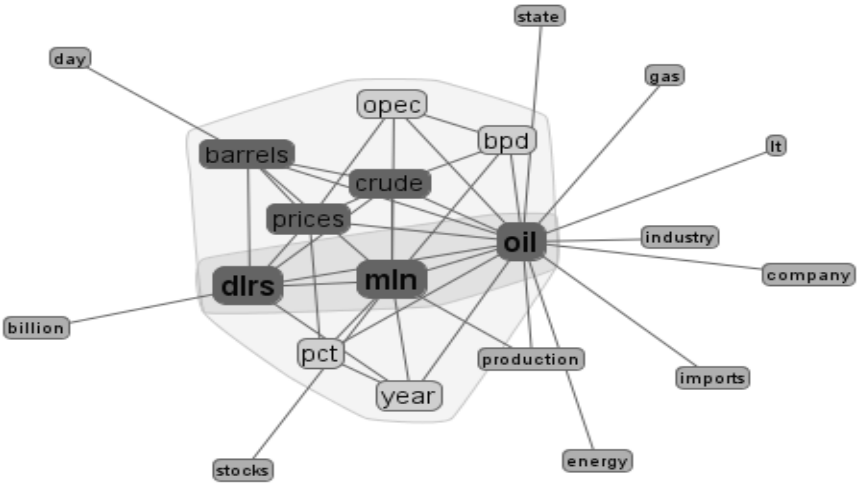

Fig. 8. Txt2vz diagram for Reuters category "crude"

The key topics words are identified as having the largest number of links (‘dlrs’, ‘mln’ and ‘oil’) and are located near the centre of the graph and surrounded with a shaded area.

6 Discussion and Future work

We have presented a new tool for generating word graphs, based on word frequency and co-occurrence, as means of identifying important topics in a document or text collection. There are many variables to assign when generating a graph such as the scale of dimension reduction, the maximum distance for co-occurrence calculations, number of word pairs used and the type of graph presented. We would like to spend more time evaluating the usefulness of the tool as perceived by human subjects. We would also like to investigate the feasibility of using Txt2vz as part of 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.

7 References

1. Viégas, F.B., Wattenberg, M., Tag Clouds and the Case for Vernacular Visualization, ACM Interactions, XV.4 - July/August, 2008
2. Y. Hassan-Montero, V. Herrero-Solana. Improving Tag-Clouds as Visual Infor-mation Retrieval Interfaces. InSciT2006, 2006.

3. Viégas, F.B., Wattenberg, M., van Ham, F., Kriss, J., & McKeon, M. Many Eyes: A Site for Visualization at Internet Scale. Proc. of IEEE InfoVis 2007.
4. Bateman, S., Gutwin, C., Nacenta, M.: Seeing Things in the Clouds: The Effect of Visual Features on Tag Cloud Selections. In: Proc. of the 19th ACM conference on Hypertext and Hypermedia, pp. 193–202. ACM Press, New York (2008)
5. P. Gambette and J. Véronis, "Visualising a text with a tree cloud," Proceedings of 11th IFCS Biennial Conference, pp. 561-570, 2009
6. D. Lin, "Using collocation statistics in information extraction", in Proceedings of the Seventh Message Understanding Conference (MUC-7), 1998
7. A. Veling, P. Van der Weerd. "Conceptual grouping in word co-occurrence networks." Proceedings of the IJCAI '99. Volume 2. Pages 694-699.
8. F. B. Viégas, M. Wattenberg, and J. Feinberg, "Participatory Visualization with Wordle," IEEE Transactions on Visualization and Computer Graphics, vol. 15, pp. 1137-1144, 200
9. C. Collins, S. Carpendale, and G. Penn, . "DocuBurst: Visualizing Document Content using Language Structure". Computer Graphics Forum, Proceedings of Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis '09)), 28(3): pp. 1039-1046, June, 2009
10. Reuters-21578 at http://www.daviddlewis.com/resources/testcollections/reuters21578/