A Graph Based Abstraction of Textual Concordances and Two Renderings for their Interactive Visualisation

Saturnino Luz
School of Computer Science and Statistics
Trinity College Dublin, Ireland
luzs@cs.tcd.ie

Shane Sheehan
School of Computer Science and Statistics
Trinity College Dublin, Ireland
sheehas1@tcd.ie

ABSTRACT

Concordancing, or the arranging of passages of a textual corpus in alphabetical order according to user-defined keywords, is one of the oldest and still most widely used forms of text analysis. It finds applications in areas such as lexicography, computational linguistics, translation studies and computer-assisted machine translation. Yet, the basic form of visualisation employed in the analysis of textual concordances has remained essentially the same since the keyword-in-context technique was introduced, over fifty years ago. This paper presents a generalisation of this technique as an analytical abstraction of concordances represented as undirected graphs, and then characterises keywords in terms of graph eccentricity properties. We illustrate this proposal with two distinct visual renderings: a mosaic (space-filling) display and a bi-directional hierarchical display. These displays can be used in isolation or in conjunction with traditional keyword-in-context components in an overview-plus-detail pattern, or as synchronised views. We discuss scenarios of use for these arrangements in lexicographical corpus analysis, in translation studies and in text comparison tasks.

Categories and Subject Descriptors

H.5.m. [**Information Interfaces and Presentation (e.g. HCI)**]: Miscellaneous

General Terms

Human Factors

Keywords

Information Visualisation, Text analytics, Linguistics

1. INTRODUCTION

Once confined to laborious reading and abstract study, text analysis has taken in the last decades a decidedly empirical and corpus-based character. This turn is due, of course, to the vast amounts of text available in digital format, but also to changes in theoretical perspectives. *Text analytics*, a term that describes computational and statistical methods for the analysis of textual resources, has become part of the vocabulary of language researchers and, increasingly, of end users of this technology [8]. Text analytics comprises not only automated methods such as machine learning and data mining, but also interactive visualisation methods that often enhance and complement the former. Examples of such text visualisation methods range from the popular "word clouds" and other forms of "vernacular" visualisation [8], to full-fledged interactive systems [9].

Text visualisation, however, has a longer history. Arranging paralell fragments of text in some order for comparison and study dates from antiquity. The advent of computers enabled the systematic creation of such *concordances* through the "keyword-in-context" (KWIC) indexing technique, first proposed by Hans Peter Luhn in the 50's. In visual representation terms [1], KWIC employs position and sometimes colour to structure text samples around a word of interest (the keyword) and its ordered left and right *contexts*. A simplified example of this type of representation is shown in Figure 1.

```
    is invisible to the naked eye. From egg to  egg
simply invisible to the naked eye. It crawled  without a
      sort out with the naked eye the blur of bodies moving
   diagnose it with the naked eye, and there are two
        at him with her naked eye, almost with curiosity.
```

Figure 1: KWIC display for the word "eye"

This form of visualisation, enhanced in interactive systems by features such as search, context sorting and statistical analysis, is still widely used by corpus linguists, lexicographers, translators and others [5]. Recently, an alternative form of visual encoding of concordances called Word Tree [9] has been proposed. It displays alternatively the left or the right context of a concordance as a tree where words are vertices linked in textual order and scaled in size according to their frequencies.

While the Word Tree can save screen space and thus present a better overview of the concordance (which often comprises hundreds or even thousands of lines of text), its underlying data abstraction, a prefix tree, limits the display to half of the text (the keyword plus the left or right context). This prevents the user from reading the full sentences in which the keyword appears. For certain corpus linguistics tasks, such as detection of phrases that span left and right contexts (e.g. as in the expression "run the whole *gamut* of ..."),

AVI'14, May 27–29 2014, Como, Italy

http://dx.doi.org/10.1145/2598153.2598187.

Luz, S., and Sheehan, S. A graph based abstraction of textual concordances and two renderings for their interactive visualisation. In *Proceedings of the International Working Conference on Advanced Visual Interfaces* (New York, NY, USA, 2014), AVI '14, ACM. in press.

frequency information for words occurring on each context is usually more useful to the analyst than the linear structure of a single context [7]. Another shortcoming of Word Trees is that a new prefix tree needs to be generated each time a new context is displayed, causing a loss of visual continuity.

In this paper we address these issues by proposing a unified data abstraction based on graph eccentricity and describe a path to implementation of concordance visualisation tools in terms of the data state framework [2]. We illustrate this by implementations of an interactive mosaic (space-filling) display, a bi-directional hierarchical display and a composite mosaic+KWIC display, which enable familiar information visualisation patterns such as overview+detail and synchronised views.

2. GRAPH REPRESENTATION

A KWIC indexing system works by circularly shifting each element of a set of ordered lines of text. For concordancing, the presentation is such that the keyword is placed at the centre of the line. We shall designate the keyword by k and its left and right contexts by $L = (l_1, \ldots, l_n)$ and $R = (r_1, \ldots, r_n)$, respectively, where l_i (respectively, r_i) denotes a word i positions to the left (right) of k. The index can then be represented by a set $\mathcal{C}$ of triples of the form $C = (L, k, R)$.

This structure encodes a high degree of redundancy in that it disregards the fact that for a given position, many different word occurrences (*tokens*) across the concordance lines are simply instances of the same word (*types*). This is illustrated by the words "naked" and "the", respectively l_1 and l_2 in the concordance shown in Figure 1. Since according to Zipf's Law a small number of types tends to dominate the distribution of tokens at a particular position, the bulk of the data in $\mathcal{C}$ will consist of such repetitions.

One can devise a more economical representation by exploiting the linear structure of C. The approach we propose does this by representing the concordance set as a graph, where vertices correspond to word types and the linear order is encoded by the edges, as follows.

Definition 1. A *concordance graph* is a quadruple $\mathcal{G} = (V, E, V_l, E_l)$ where V is a set of vertices, $E \subseteq V \times V$ is a set of edges (v_s, v_t) connecting vertices, $V_l : V \to Types$ is a labelling of vertices with words and $E_l : E \to \mathbb{R}$ is a labelling of edges with word frequency information.

The word frequency labels in E_l are assumed to indicate the number of concordance lines between the two ends of the edges. A concordance graph can be built through an algorithm that takes a KWIC index $\mathcal{C}$ (encoded, say, as a tabular structure) as input and

1. cycles though each lexicographically sorted column, starting from the centre (corresponding to k, with index $i = 0$) and expanding over L and R,
2. creates a vertex $v_{i,j}$ for each type (in row j of column i), labelling the vertex with the appropriate string,
3. recursively connects each vertex to the next column's vertices $v_{i+1,m}$, labelling edges according to the number of strings $v_{i,j}v_{i+1,m}$ in the concordance,
4. and, finally, creates edges linking each vertex v_n^l for each row in the leftmost column of C to the corresponding vertices v_n^r for the rightmost column. We will refer to such edges as *contextual edges*.

Figure 2 shows an example of concordance graph for the fragment seen in Figure 1 with word count labelling . Note that the edges that connect the left to the rightmost vertices guarantee that the entire set of concordance lines going through any vertex $v_{i>0}$ is retrievable by traversing the concordance graph starting from v_i, which is not possible in a concatenation of Word Trees.

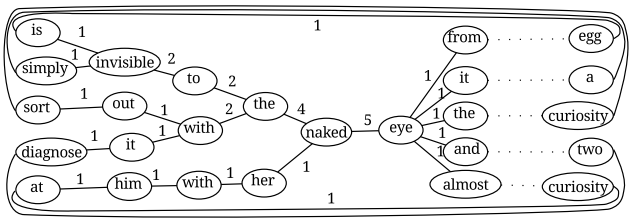

Figure 2: Sample concordance graph for the word "eye".

If we define graph distance $d(v, u)$ as the minimum length of the paths connecting vertex v to u in concordance graph $\mathcal{G}$ and an operation $P(\mathcal{G})$ which removes all contextual edges (v_n^l, v_n^r) from $\mathcal{G}$, then we can retrieve the keyword vertex through its eccentric property.

Definition 2. The *eccentricity* $\epsilon(v)$ of a vertex v in a concordance graph is defined as $\epsilon(v) = \max_{u \in V \setminus \{v\}} d(v, u)$. The minimum graph eccentricity $(\min_{v \in V} \epsilon(v))$ is the *graph radius*.

Given a concordance graph $\mathcal{G}$, the keyword k is the label $V_l(v_k)$ of the vertex v_k whose eccentricity $\epsilon(v_k)$ is equal to the graph radius of $P(\mathcal{G})$. The above described graph construction algorithm guarantees that this vertex is unique and corresponds to k in the KWIC representation. This can be seen from the fact that every vertex in $P(\mathcal{G})$ can reached from every other vertex through v_k.

3. VISUALISATIONS

To illustrate a use of the concordance graphs we have created two basic visual renderings. The first rendering is a mosaic display and the second a bi-directional hierarchical display.

We built these visualisations by extending an existing corpus browser. This corpus browser allows the user specify a keyword and returns the keyword with both the left and right contexts, the result is then displayed as a keyword-in-context view. The plugin architecture implemented for building the new interactive visualisations allows access to the results of the keyword search and other aggregate corpus information (e.g. word frequency within the corpus).

Using Chi's and Riedl's reference (data state) model [2] we show the data states and operators which create our visualisations. This model is shown in Figure 3. We chose to create the visualisations using the Prefuse library as it's software architecture is also based on this reference model [3].

At the visual abstraction level we have two data sets, one for each visualisation. The the data set used to construct the mosaic view is structured as a collection of vectors containing word objects. These vectors are ordered so that if vector x contains the keyword then vector $x + 1$ contains all

Luz, S., and Sheehan, S. A graph based abstraction of textual concordances and two renderings for their interactive visualisation. In *Proceedings of the International Working Conference on Advanced Visual Interfaces* (New York, NY, USA, 2014), AVI '14, ACM. in press.

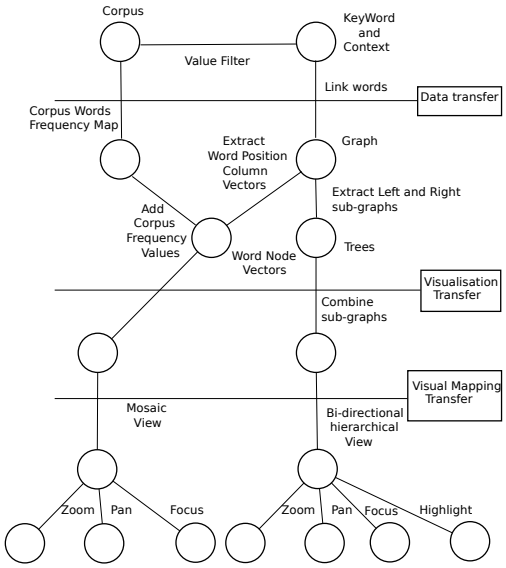

Figure 3: Concordance visualisation reference model diagram

words which occur one position to the right of the keyword (in the corpus) and vector $x-1$ all words one position to the left. These words (word objects) consist of the word token, a value representing the frequency with which this word has occurred at this position and the words frequency of occurrence in the entire corpus, as extracted from the concordance graph.

The mapping of this dataset to the mosaic view entails laying out the vectors as columns in a grid, creating and scaling rectangles within the columns to represent the words, and trying to assigning a random unique color to distinct words. This design is visually similar to the temporal mosaic [6].

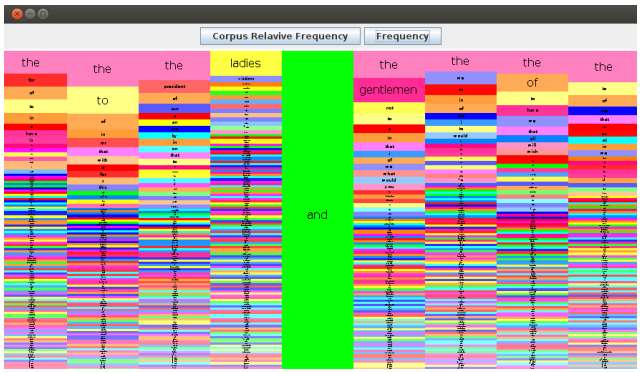

Figure 4: Mosaic scaled on word frequency

Two different scaling schemes were explored. One representation (Figure 4) shows the word rectangles with vertical height scaled according the frequency of the word in the column. The other (Figure 5) scales the height of the rectangles on their column frequency relative to the word frequency in the corpus. This relative frequency representation has the effect of reducing the size of words which have high frequency in the corpus and increasing words of lower frequency (e.g. comparing figures 4 and 5 the word "the" has been reduced and the word "review" has increased in the corpus relative frequency view). In the relative frequency representation we have also reduced the size of rectangles which have relative frequencies below a certain threshold, this gives more screen space to the words which more often occur with the keyword.

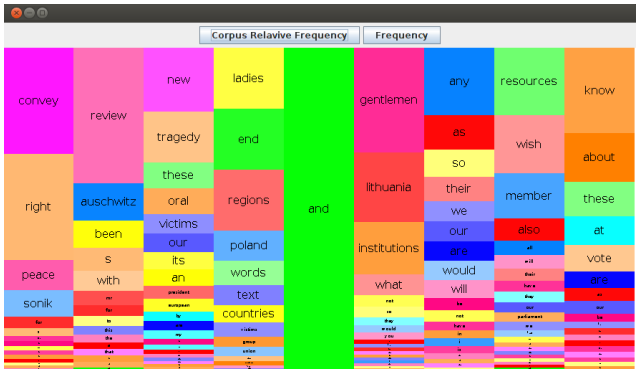

Figure 5: Mosaic scaled on word frequency relative to corpus word frequency

The data set used to construct the bi-directional hierarchical display is structured as a concordance graph, from which trees corresponding to the left and right contexts are extracted for visualisation transfer (graph layout operations). Once layouts for the contexts have been defined, the subgraphs are then recombined for presentation. The visual mapping operation scales the fonts used to display the label of each vertex (a word) on the graph proportionally to the maximum frequency label of the edges that are incident on it. This might have resulted in loss of positional information, as happens in the Word Tree [9]. We, however, chose to preserve position as one of the main visual variables [1] in order to retain some resemblance with the KWIC presentation.

Figure 6 illustrates the hierarchical display for a subset of the concordance for the keyword "eye", focusing on the expression "naked eye". The left patterning for this expression, including the semantic prosody of difficulty ("invisible", "barely visible") [7] is evident in this display. However, when choosing a word on the left (e.g. the word "invisible") and traversing the graph from left to right, many of the sub-trees to the right of the keyword will not be part of sentences containing the chosen word. Using the contextual edges, however, we can reconnect the two contexts by highlighting the words that appear in the same sentences as the chosen word (yellow highlights in Figure 6) while preserving continuity.

The advantage of the mosaic and hierarchical visualisations, over the traditional keyword-in-context view, is that they provide an overview of the context and can provide an at-a-glance review of properties such as word frequency. However the mosaic view loses some of the detail of the traditional view, that is, the underlying sentence structure.

We used juxtaposed views [4], a design pattern of composite visualization views, to make use of the advantages of both the traditional and mosaic concordance views. This design gives the user both the overview and detail. Since the data is implicitly linked, interaction with either view can affect the other. We demonstrate this by a applying a filter interaction on the juxtaposed views (Figure 7). Selecting

Luz, S., and Sheehan, S. A graph based abstraction of textual concordances and two renderings for their interactive visualisation. In *Proceedings of the International Working Conference on Advanced Visual Interfaces* (New York, NY, USA, 2014), AVI '14, ACM. in press.

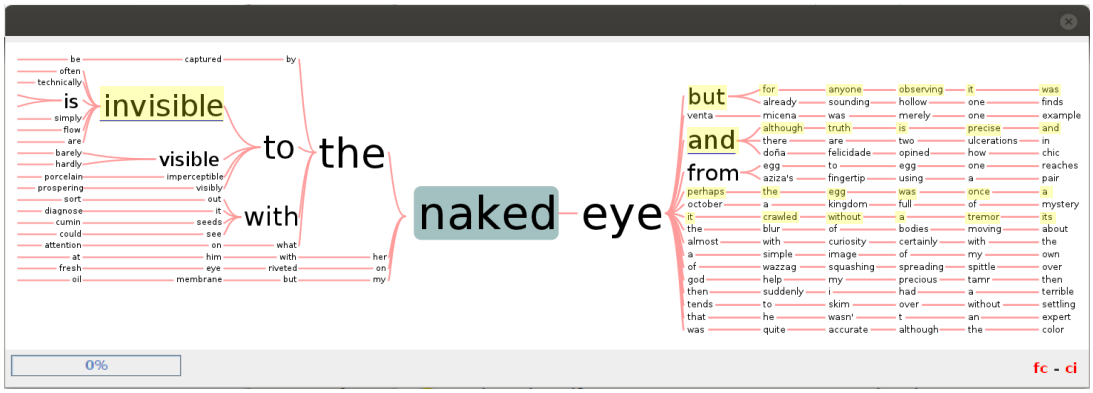

Figure 6: Bi-directional hierarchical view of a concordance for the compound "naked+eye"

a word on the mosaic view filters the traditional keyword-in-context view, such that only sentences which contain the highlighted word in the selected column are visible.

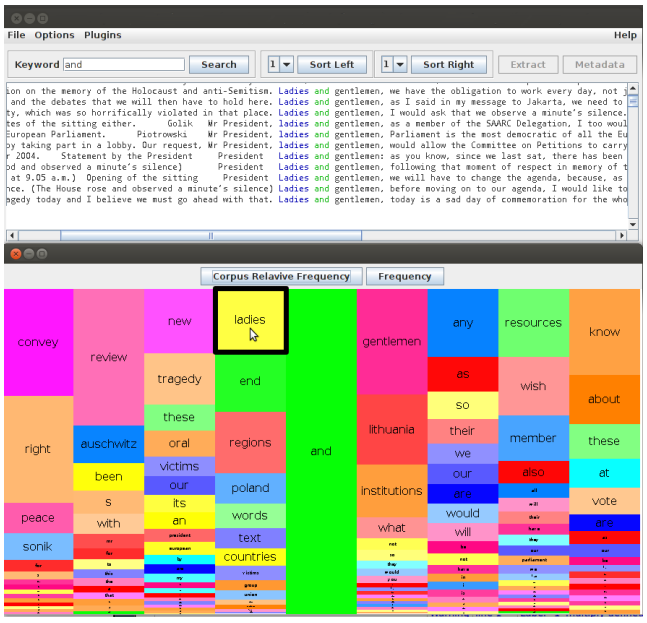

Figure 7: Juxtaposed views filtering example

4. CONCLUSION

We introduced concordance graphs, a generalisation of the keyword-in-context method, in which the keyword is encoded in terms of vertex eccentricity and sentence structure is preserved from the the left to the right contexts of the keyword. This opened up visualisation possibilities not available to recent tree-based implementations of concordance displays. We illustrated these possibilities with mosaic and hierarchical visualisations.

Both visualisations have advantages but also some shortcomings when compared to the traditional KWIC tables. By rescaling the nodes proportionally to frequency, the hierarchical displays emphasises collocation patterns, as required in lexicography analysis. However, the further away from the keyword a word is, the smaller it will appear making reading difficult for large concordances. The mosaic display employs a space-filling technique that partially overcomes this limitation at the cost of sentence structure.

Preliminary qualitative feedback from professional translation studies researchers suggests that the visualisations presented can be productively used in corpus analysis and that the above discussed issues can be addressed with composite visualisations. We plan to conduct a controlled study to quantify the potential advantages these techniques.

5. ACKNOWLEDGMENTS

This study was funded by the Science Foundation Ireland through the CNGL grant number 07/CE/1142.

6. REFERENCES

[1] J. Bertin. *Graphics and graphic information-processing*. de Gruyter, 1981.

[2] E. H. Chi and J. T. Riedl. An operator interaction framework for visualization systems. In *Procs. of the IEEE Symposium on Information Visualization*, pages 63–70, 1998.

[3] J. Heer, S. K. Card, and J. A. Landay. prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430. ACM, 2005.

[4] W. Javed and N. Elmqvist. Exploring the design space of composite visualization. In *Pacific Visualization Symposium (PacificVis), 2012 IEEE*, pages 1–8, 2012.

[5] S. Luz. Web-based corpus software. In *Corpus-based Translation Studies – Research and Applications*, chapter 5, pages 124–149. Continuum, 2011.

[6] S. Luz and M. Masoodian. Visualisation of parallel data streams with temporal mosaics. In *Procs. of the 11th International Conference on Information Visualisation*, pages 196–202, Zurich, 2007. IEEE.

[7] J. Sinclair. *Reading Concordances: An Introduction*. Longman Publishing Group, 2003.

[8] F. Viégas and M. Wattenberg. Tag clouds and the case for vernacular visualization. *Interactions*, 15(4):49–52, 2008.

[9] M. Wattenberg and F. B. Viégas. The word tree, an interactive visual concordance. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1221–1228, 2008.

Luz, S., and Sheehan, S. A graph based abstraction of textual concordances and two renderings for their interactive visualisation. In *Proceedings of the International Working Conference on Advanced Visual Interfaces* (New York, NY, USA, 2014), AVI '14, ACM. in press.