Journal of Northwest Semitic Languages 35/1 (2009), pp. 75-95

J H Kroeze (North-West University, Vaal Triangle Campus), Theo J D Bothma, Machdel C Matthee, Ockert C Kruger & Jan C W Kroeze (University of Pretoria)

INTERACTIVE NETWORK GRAPHS OF BIBLICAL HEBREW DATA[1]

ABSTRACT

Knowledge workers, including Biblical Hebrew computational linguists, should look into the possibilities offered by graphical visualisation techniques to allow explorative investigation of available linguistic data, since this may prompt new hypotheses, which may then be examined in more traditional, empirical ways. This article experiments with two-dimensional and three-dimensional implementations of interactive network graphs to enable dynamic, "what-if" investigations, using semantic-role data from Genesis 1:1-2:3, marked up in XML.

1. INTRODUCTION

It is well known that information technology changes the working culture and structure of organisations (see Du Plooy 1998:12-23). A similar phenomenon can be observed in humanities computing areas. For example, while electronic dictionaries initially mimicked printed lexicographical styles, in the course of time researchers realised that the available technology actually facilitates many more ways to render such data. Slowly but surely this has led to innovative ways of organising dictionary content (Manning *et al.* 2001:136). Similarly, this article explores new renderings of linguistic data, which are made possible by visualising XML-based clausal data.

The article briefly discusses the need for, and purpose of, visualisation of linguistic data. After a definition of visualisation is given, one-, two- and three-dimensional approaches are discussed. An XML database is identified as suitable technology to store the underlying data. The relationship between visualisation and text mining is investigated.

1 This article is a revised version of an (unpublished) paper read at the AIBI VIII Conference in San Lorenzo de El Escorial, Spain, June 2008, "Designing an interactive network graph of modular linguistic data in an XML database of Biblical Hebrew." The conceptual part of the article is based on selected sections of Chapter 7 of a PhD thesis (Kroeze 2008). The empirical work, however, was done after completion of the thesis.

Finally, an interactive network graph is designed and implemented in two-dimensional (2D) and three-dimensional (3D) versions visualising networks of semantic roles with a view to enabling a researcher to interactively explore Biblical Hebrew data for interesting patterns.

2. *THE NEED FOR VISUALISATION IN LINGUISTIC INFORMATION SYSTEMS*

Classic information retrieval is search-dominated and only effective if the user already has a good idea about the problem space and knows what to look for. "Such interfaces are ineffective for information needs such as exploring a concept" (Manning *et al.* 2001:136). Visualisation provides an alternative that could fill this gap.

According to Freedman (2006:24), the visual presentation of data is necessary

- to help the vast majority of visual learners to pinpoint suspected correlations in data;
- to make it easier to see relationships in large and complex sets of data;
- to facilitate exploration of data to discover unexpected patterns.

Using visualisation techniques for linguistic software may also meet the educational needs of users who are not highly trained linguists, for example, dictionary users, and in the case of the Hebrew Bible, ministers and preachers. Providing affordable, customisable access to linguistic sources may even help to keep endangered languages alive (cf. Manning *et al.* 2001:135, 137). Biblical Hebrew scholars should also make use of these technologies to protect knowledge and promote research on this ancient language.

3. *A DEFINITION OF VISUALISATION*

Visualisation is a graphical display of data that facilitates advanced information retrieval and exploration. It is an innovative way of representing data, one of the basic ventures of humanities computing (Neyt 2006:2-5). Visualisation is usually studied by multimedia researchers and, therefore, forms part of Information Science. However, the selection and manipulation of the subset of the data that will be displayed are based on algorithms that are researched by computer scientists. Computer Science makes another contribution in terms of the programming of graphical interfaces. Visualisation also has links to Information Systems, such as the facilitation of text mining, as well as the study of human computer interaction, which is used to evaluate visualisation products and to test their usability.

4. *FORMATTING AND PREPARING THE UNDERLYING DATA*

Text analysis is not only a process of analytic decomposition, but also re-composition, both of which phases take place in a circular or spiral-like fashion (cf. Sinclair 2003:181). A complete tool should, therefore, also address the synthetic processes in reproducing the original text or creating new texts such as annotated versions or reports. The mark-up language XML provides suitable technology to prepare the underlying linguistic data that should be visualised because of its extensibility and customisability (Walsh & Muellner 1999:4). Combining the XML data file(s) with suitable style sheets will enable re-composition of the original texts in various forms as may be required.

An XML file itself, however, is not a user-friendly document to read. Even though it is text-based and does not use complex programming syntax, the tags obscure the underlying text. The reader has to understand the basic hierarchy and schema to make any sense of the text file. Even if one knows the structure, it is not nearly as easy to read as a database table. XML is essentially a one-dimensional stream of text. Even the use of indentation to highlight the hierarchy of tags is "just barely two dimensional" (Bradley 2003:199). However, many features may be recorded as mark-up that may be used by other programs to produce suitable visualisations. XML's weakness is therefore also its strength: there exists one master copy of the marked-up text, which can be formatted in many different ways according to the needs and wishes of researchers and users (Flynn 2002:57).

Although an XML file uses relatively little space, because it is text-based, it could still become too big. When it becomes so large that it slows down the parsing speed to an unacceptable level or that the data becomes too dense to be shown on a computer screen, the parsing algorithm should be adjusted to parse subsections of the database as required (cf. Manning *et al.* 2001:141). The XML database may, for example, be broken down into logical and physical chunks so that different books or sections may be stored in separate files, which are defined as entities in the original document and inserted into it when referenced (cf. Walsh & Muellner 1999:30).

While the size of the XML databank was not a problem in the limited experiments for this paper, the problem of handling embedded clauses could complicate matters. The problem was addressed by presenting these clauses as separate entries linked by means of the clause IDs. This solution is similar to the approach followed by Manning *et al.* (2001:141) of handling sub-entries in a dictionary. It facilitates not only a simpler XML structure but also easier visualisation strategies. The data of the Biblical Hebrew text was marked up on various linguistic modules and captured in an XML file with an underlying three-dimensional structure, which may be called a *clause cube* (a specialised version of a data cube).

Although more than one visualisation or even a customisable visualisation of the same XML-based data would facilitate more interpretations, one has to acknowledge that even these are limited to the theory and interpretation of the persons who tagged the data and created the software to access and analyse it (cf. Neyt 2006:7).

5. *ONE-, TWO- AND THREE-DIMENSIONAL REPRESENTATIONS*

The various approaches towards visualisation may be categorised according to the number of dimensions represented. A text-based representation of the results of a text-mining exercise may be regarded as a one-dimensional rendering of the data. It is a stream-based, non-graphical presentation that only qualifies for the term visualisation in the widest sense of the word. The results of the analyses of semantic frames presented in a textbox may be regarded as a very simple one-dimensional display of the results of a rather complex algorithmic phase (cf. Kroeze 2007). The user is required to read linearly in order to access the information it contains. To improve the usability of the interface, various fonts, colours and backgrounds could be used to highlight the various linguistic layers (cf. Neyt 2006:2-7).

Another, more advanced, text-based visualisation tool could be built using some of the guidelines given by Sinclair (2003:178-180). His approach is to conserve the original text as a basic interface in order not to alienate users who do not have advanced computer-literacy skills. Because literary critics are most familiar with printed texts, a visualisation tool should not banish the original texts, but rather exploit them as a user-friendly point of departure. This can be implemented by showing "the text as a readable whole" on the screen. Although a search function is very simple, it still is one of the most powerful functionalities of electronic text: the user could, for example, click on a word in order to "jump" to the next instance of the same word. The frequencies of word occurrences could be indicated by various shades of a specific colour. Pop-up boxes could be used to unveil related information when the user lets the mouse hover over a specific word or phrase. Words could also be used as multi-dimensional links by using menu options to trigger available analysis options for selected words.

A table is a relatively simple, but very efficient, two-dimensional, non-graphical rendering of data. A clause can be analysed very effectively using columns to represent words and rows to show the various linguistic modules. Using a table with five columns and four rows (see Figure 1),[2] Petersen (2004) analyses the sentence "The door was blue" on the levels of word, phrase and clause.

	1	2	3	4
Word	w: 10001 surface: The psp: article	w: 10002 surface: door psp: noun	w: 10003 surface: was psp: verb	w: 10004 surface: blue psp: adjective
Phrase	p: 10005 phr_type: NP		p. 10006 phr_type: VP	p. 10007 phr_type: AP
Clause	c: 10008			

Figure 1. A two-dimensional visualisation of a sentence analysis on various levels (Petersen 2004).

A similar approach is followed by Kroeze (2002) using linguistic data from Jonah. The cells on the various levels could "span" different parts of the primary data (original text) to reflect the unique partitioning on the different layers (Witt 2005:76-77; compare the spanning on the phrase and clause levels in Figure 1). However, spanning is difficult to implement in a data cube and will not be discussed in further detail in this article.[3]

A style sheet, such as a css file (cascading style sheet), used to show the contents of an XML clause cube as a series of interlinear tables in a web browser, is a mechanism to implement a two-dimensional

2 Compare Petersen (1999:14) for a more extensive example including a relative clause.

3 Compare Witt (2005), who suggests that overlapping structures be annotated in separate XML documents using heterogeneous tag sets, but linked by using the basic text as primary data and implicit links. Other solutions for this complex problem are CONCUR (available in SGML but not in XML), milestone elements ("empty elements which mark the boundaries between elements, in a non-nesting structure"), fragmentation and nesting, virtual joins, redundant encoding and standoff annotation (Witt 2005:58-59).

representation of the XML data cube.[4] To show all of the data, a series of disconnected tables has to be used. Although a table uses two dimensions, it is essentially still text-based and does not utilise more advanced information technologies.

Ideally, one should explore the possibility of three- or multi-dimensional visualisations to render inherently multi-dimensional data, such as linguistic analyses. "It is our conjecture that linguistic meaning is intrinsically and irreducibly very high dimensional" (Landauer *et al.* 2004:5214). This is also the case for the Genesis 1:1-2:3 data, used in this article, that has been captured in a three-dimensional XML data structure.

While the two-dimensional table approach, discussed above, is very limited and cannot visualise a collection of clauses in one "picture", a three-dimensional picture could show the various sentences stacked as layers in a cube. This would be very difficult or almost impossible to be represented on paper, but computer animation, on the other hand, can facilitate three-dimensional simulation much better because "it can render movement and time" (Neyt 2006:7). Movement could be used in rotation to reveal the various faces of the three-dimensional clause cube, as well as in slicing and dicing and drilling-down processes.

An interlinear two-dimensional approach would probably be sufficient for a Bible reader or user who only needs enough information to understand the Hebrew text. For the in-depth researcher, however, three-dimensional displays could be very helpful, if they could reflect the complex, multi-dimensional patterns and structures which are concealed underneath the surface structure of a stream of a one-dimensional text. "Information visualization can present multiple dimensions of information that can be extremely useful in helping an analyst quickly sift through information to find patterns, filter the data live, and drill down to more meaningful result sets. These result sets can subsequently be exported via XML to other analytical packages" (Freedman 2006:24).

---

4 Internet Explorer 6 is not able to use an XML style sheet to directly show the XML data in table format, but Opera7 could be used instead. An alternative approach could be to use a detour by means of XML data binding in MSIE 5.0. A separate HTML file is created that contains a code to incorporate the XML data and bind it to an HTML table (Rob & Coronel 2009:598).

6. *USING VISUALISATION FOR TEXT-MINING PURPOSES*

Visualisation could either be used before, during or after a data-mining or text-mining venture. According to Thuraisingham (2002:87-89, 279), the following scenarios are possible:

- After using a data-mining tool to identify patterns in the data, the results are presented in a user-friendly interface using visualisation techniques;[5]
- Essential elements of the raw data are first represented visually and then explored using data-mining algorithms;
- Visualisation techniques are used to complement data mining, e.g. to fine-tune or better understand correlations and patterns already found;
- Visualisation steers data mining, for example, to dynamically change the focus of the mining process or to implement what-if scenarios.

The final option listed above is the most advanced and complex use of visualisation, since it involves visualisation in all phases of the data-mining procedure. It also provides an interactive and experimental tool for the researcher. According to Sinclair (2003:182), computer-assisted "play" or experimentation is a suitable method for humanities computing of literary texts, and visualisation may be used to implement this.[6]

Because play "thrives on improvisation and imagination," such a tentative, investigational research method complements the more rigorous and purposive research activities such as hypothesis testing (Sinclair 2003:181). Experimental exploration may stimulate new ideas and may be used to formulate new hypotheses, which may then be examined in more traditional, empirical ways. Sinclair therefore

---

5 According to Freedman (2006:24), information visualisation is "an interface that sits atop the data mining reporting program," bringing to the fore complex patterns hidden in multi-dimensional data.

6 Computer-assisted reading and text synthesis are other ways of making the use of computers more acceptable to humanities scholars. Computer-assisted reading could be regarded as a more advanced visualisation technique than an interlinear approach, which is, however, still text-based. Computer-assisted text synthesis could be used in the report function of a visualisation tool to produce suitable outputs of research results, to reproduce the original text (without mark-up and analysis), or to create an amended text according to the end users' requirements (cf. Sinclair 2003).

advocates "a hybrid model of formal, analytical functions and interpretive, exploratory functions."[7] In this way, the visualisation tool may be used to support the formulation of a mental model (Bradley 2003:185). "It is, of course, possible to go to a literary text armed with a hypothesis, but we do better to go to it with a hunch borne of our collective musings – a sneaking suspicion that looking at it *this* way will turn up something interesting. Or better still, we could go to it with a machine that is ready to reorganize that text in a thousand different ways instantly" (Ramsay 2003:171).

7. *POSSIBLE APPLICATIONS TO BH LINGUISTIC DATA*

Bradley (2003:197-199) discusses feature structures and topic maps as examples of electronic tools that support the creation of mental models for literary analysis. A feature structure is a logical organisation of concepts identified in a text. The concepts are marked in the original texts by means of tags. Related concepts are linked in an intermediary document and these are then reorganised to form a final conceptual model as output. This could be visualised as a mind map, which is still more logically than graphically oriented. In a grammatical project a similar approach could be used, for example, to compile a concept map of semantic roles and their relations to the various predication types. A topic map is more graphical, since it contains a spatial element showing links and associations between selected topics in texts. It may, for example, draw a picture with the help of a visualisation tool linking these topics to the texts where they appear (cf. Kroeze, Bothma, Matthee & Kroeze 2008).

A graphical alternative to a topic map is a network graph. Manning *et al.* (2001:139) give an example of a graphical rendering of related lexemes in a dictionary. These words form a graph with nodes and links. A similar approach is suggested here to visualise the networks of semantic roles and frames. It could, for example, show all the semantic frames in Genesis 1:1-2:3 containing purpose as one of its elements. The network could be constituted by connecting identical predicate types (action, position, process or state) and frames that share the same semantic roles. The semantic role that is selected as focus point (for

---

7 According to Keller et al. (2006:46), a combination of text and graphics is ideal for information processing, since it uses both the verbal and visual memory systems.

example, by clicking on it in a drop-down list) could be highlighted.[8] This should bring together all the essential information about the semantic frames in which purpose appears and may help the researcher to either confirm his/her theory or to falsify it, or to reveal new patterns and prompt new hypotheses. If only a formatted text document is available, the researcher first has to search for all frames containing purpose and then analyse this subset manually to reach the same point. More colours could be used by highlighting the various semantic roles and predication types in different colours and by using different shades of one colour to indicate the frequencies of each frame. A drop-down list could be added to provide hyperlinks to the clause instances of each frame. Like a drop-down list showing an alphabetical list of lexemes in a dictionary, this may provide "concreteness" or "tangibility" to the interface by providing access for the user to "what lies among the electrons beyond the screen" (Manning *et al.* 2001:139). When the user sets the focus on another semantic role, the graphical interface should change accordingly. One or more multi-line textbox panes may be added to show more detail on the frames or roles, for example, a relevant extract of a text file (cf. ibid.). A spring algorithm[9] may be used to enable the user to interactively move the nodes on the screen – ideally, the algorithm should be intelligent enough to readjust the information accordingly. Such an "opportunistic exploration of networks" could support the researcher's data-mining activities by revealing new patterns (cf. Manning *et al.* 2001:142, 147).

One of the requirements of a proper rendering of syntactic structures of Biblical Hebrew (BH) is that it should be pictorial, that is "clearly and concisely diagrammed" (Andersen & Forbes 2003:44). The authors wish to apply this principle to the clausal semantics module and trust that their

8 This may be regarded as a "focus + context strategy, which shows details at a focus point chosen by the user while still keeping the context or overall overview" as opposed to an "overview + detail strategy" that works the other way round (Eden 2005:63).

9 Huang *et al.* (1998:643) developed a modified spring algorithm that facilitates spatial manipulation of nodes on a graph to allow users to explore revealed connections interactively and incrementally. The lexicographical graph, which is referred to above, uses this technology to implement iterative updating, "which means that users can drag nodes across the screen, and the algorithm will cause other nodes to flee out of the way, while words related to another word are dragged along" (Manning et al. 2001:142).

proposal will make a contribution to the field of BH linguistic information systems by moving beyond the limits of single clauses.

Figure 2 suggests a possible design for the implementation of such a clear, concise, pictorial diagram. It assumes that a linguistics researcher would typically want to study semantic frames containing the semantic role of purpose. When he/she selects this role, the node should be highlighted and lines should connect the frames in which it appears within the dataset. The graph should show that purpose occurs in five frames. Each frame should again be linked to all other semantic roles occurring in them. Frame 9, for example, consists of an action predication, in combination with agent, product and purpose elements. Purpose also appears in frames 4, 31, 34 and 35. This information may be used, for example, to check definitions of semantic roles. In this example a filter should have been used to show only those networks that contain purpose in their frames. Such a graphical tool could provide a more interactive, visual way to get to the same results reached in Kroeze (2007:72).

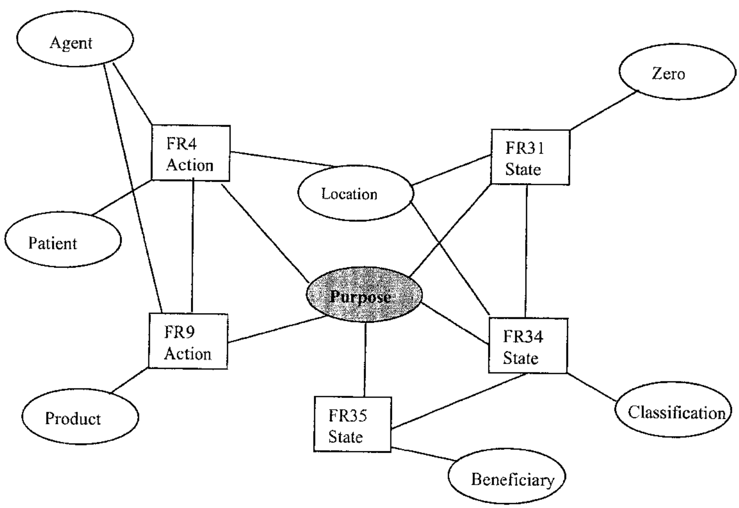

Figure 2. A proposal for a graphical visualisation of a network of semantic frames in Genesis 1:1-2:3 containing purpose as one of their elements (based on an idea for lexical visualisation by Manning *et al.* 2001:139).

8. *IMPLEMENTATION AND RESULTS*

In this section two experiments with interactive network graphs are discussed (figures 3-9 have been edited for clarity and legibility). Firstly, Visual Basic 2005 was used to create a software tool that filters the linguistic data according to the user's requirements, and to draw a two-dimensional graph based on the selected data. Figure 3 shows, for example, the default screen when the user requests all data containing the semantic role of purpose. The related elements, i.e. the semantic roles (in oval shapes) and semantic frames (in numbered rectangles), are connected by lines and shown in a random format on the screen. The filtering semantic role (purpose) is shown in a prominent colour (red).

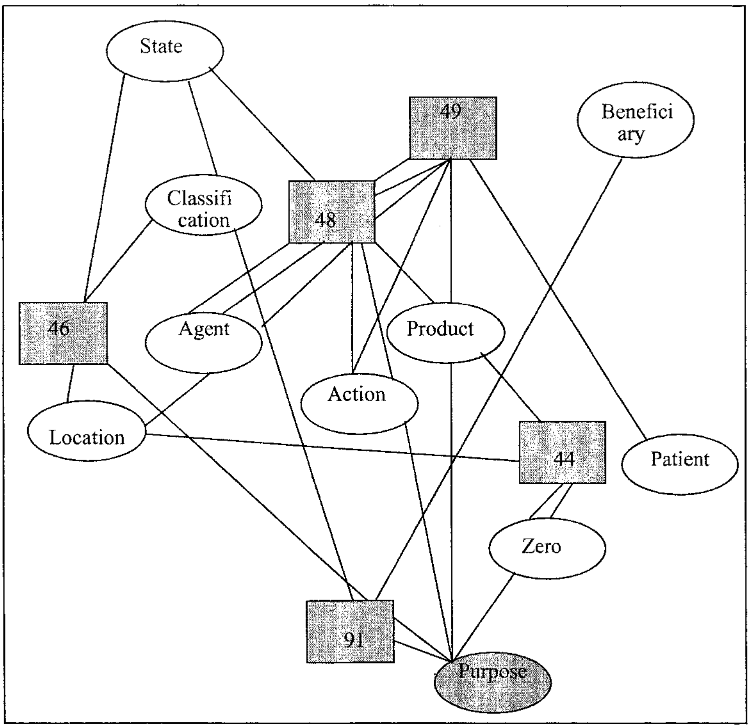

Figure 3. Default 2D network graph, implemented in Visual Basic 2005, showing a network of related semantic roles and frames, all containing purpose as one of their elements.

In order to make the information more user-friendly the tool allows one to reposition the elements on the graph by clicking and dragging the ovals representing the semantic roles and the numbered frame rectangles. Figure 4 shows how one can closely imitate the design suggested in Figure 2 above by using this facility.

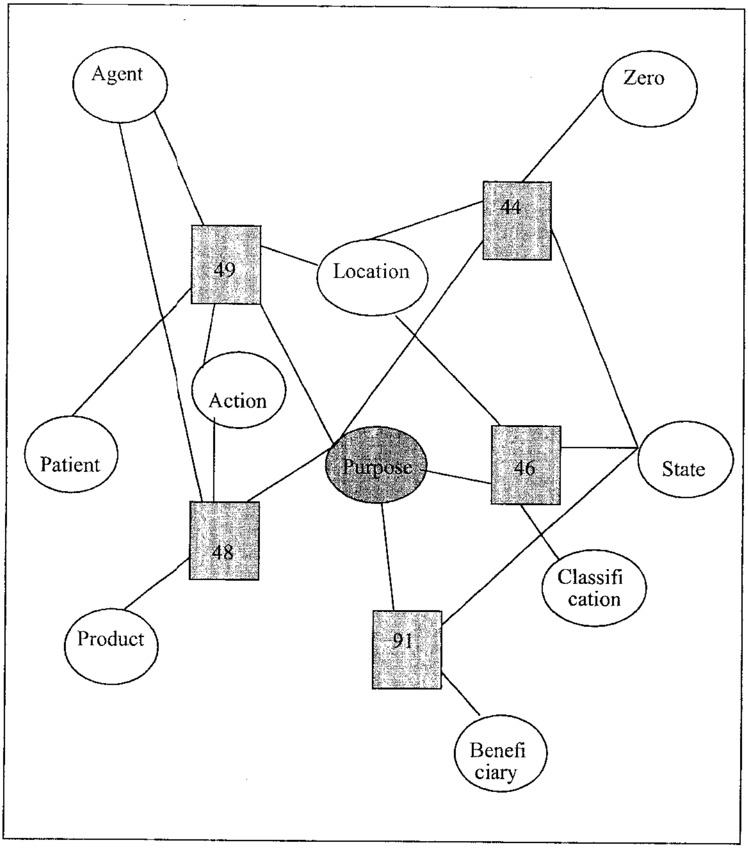

Figure 4. Repositioning the elements on the network graph manually.

The tool also provides some drill-down facilities. Figure 5 shows the information about the semantic frames that is presented in a separate pane, organised according to the numbers allocated to the frames, as well as the interactive pop-up boxes that appear when the user hovers the mouse over one of the frames in the graph.

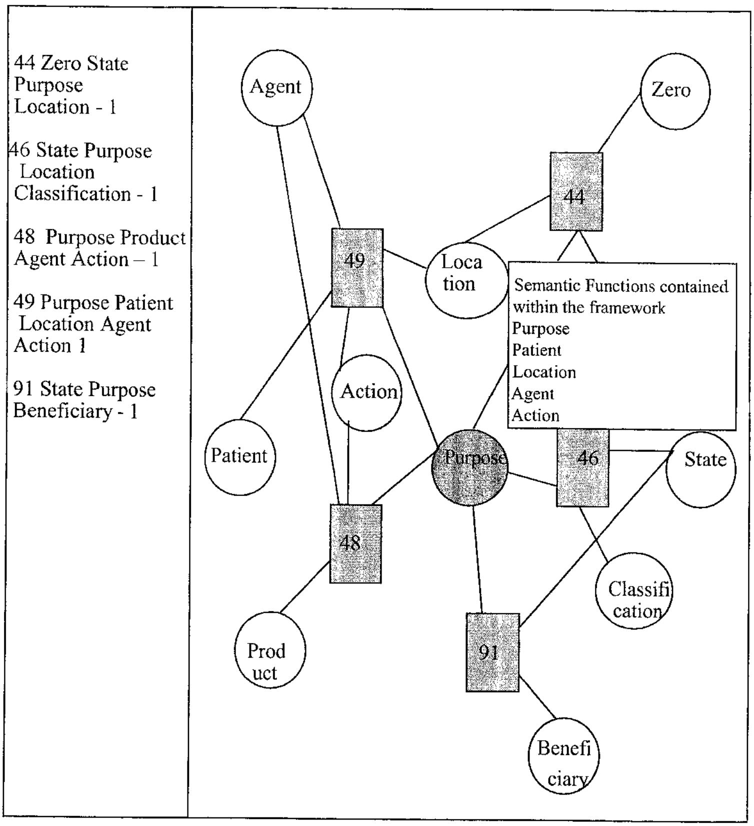

Figure 5. A separate pane and pop-up box providing drill-down facilities with detailed information regarding the constitution and frequency of relevant semantic frames.

Another facility is a pop-up box that shows the frequency of a specific semantic role in the current database, as shown in Figure 6 (indicating that purpose appears five times in Gen. 1:1-2:3).

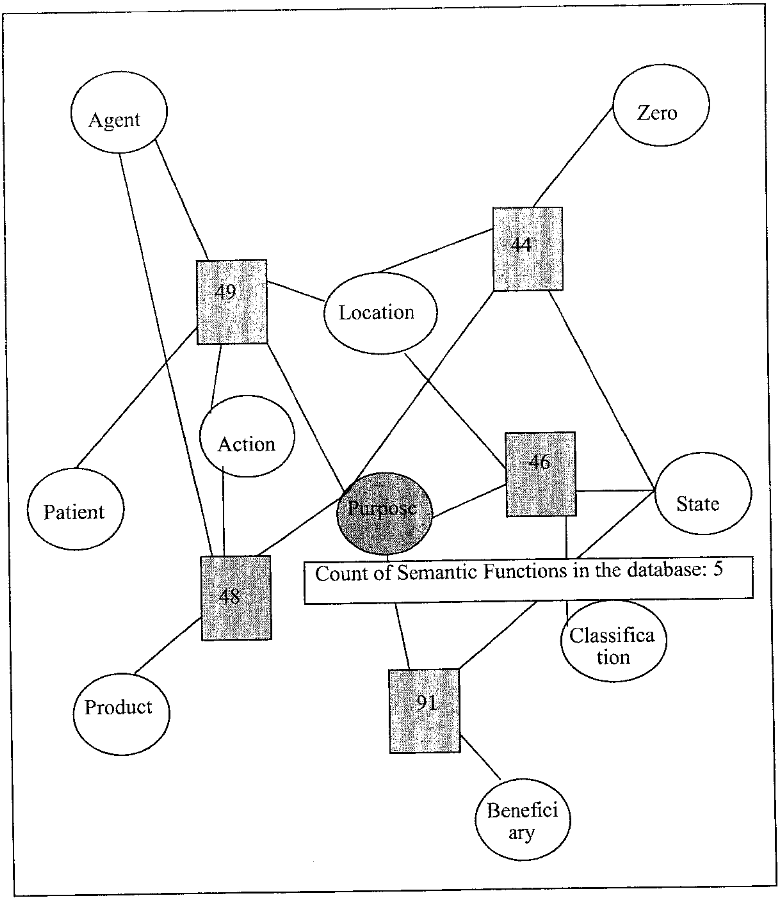

Figure 6. A pop-up box shows the frequency of a specific semantic role (purpose) when the user hovers the mouse over the name of the role.

Secondly, a 3D graph was implemented using the programming language C++. This implementation shows only the network of individual semantic roles, but can be rotated interactively to allow different perspectives on the elements. Figure 7 shows the default 3D graph when the data set is loaded, while Figure 8 shows a rotated version of the same graph. Coloured lines have been used to differentiate various semantic frames.

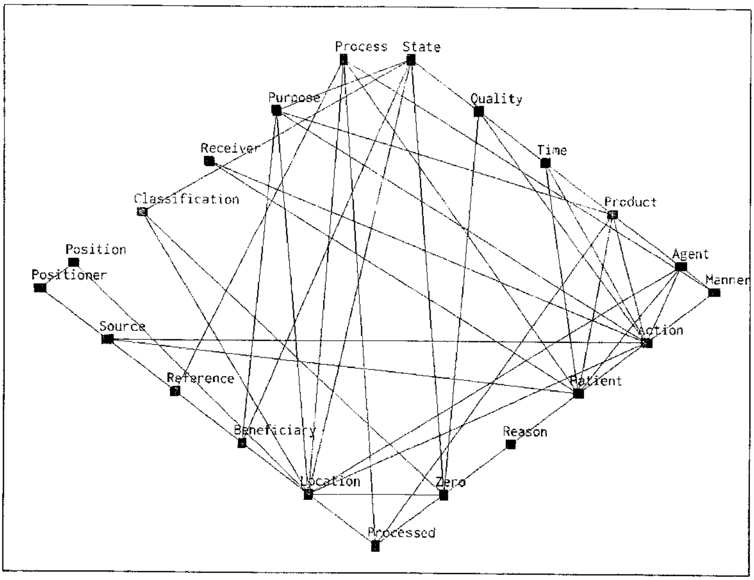

Figure 7. The default screen of a 3D interactive network graph, showing a network of semantic roles that are connected based on their collocations in the same frames.

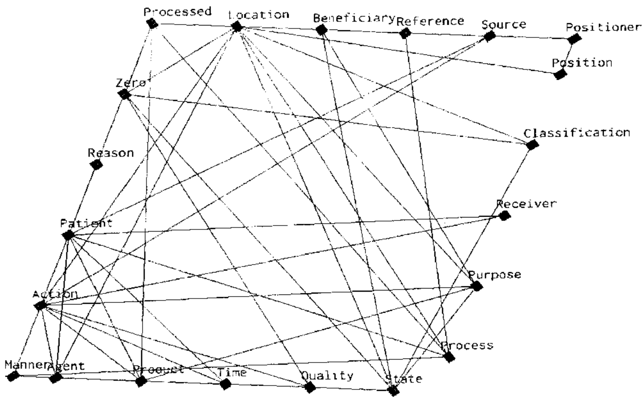

Figure 8. A rotated version of the network graph of semantic roles shown in Figure 7.

Filtering can be done by hovering with the mouse on a specific semantic role. Figure 9 is a filtered network graph showing only those semantic roles that occur in one or more semantic frames containing purpose. The name of the semantic role that is used as the filter (purpose) is highlighted in orange.

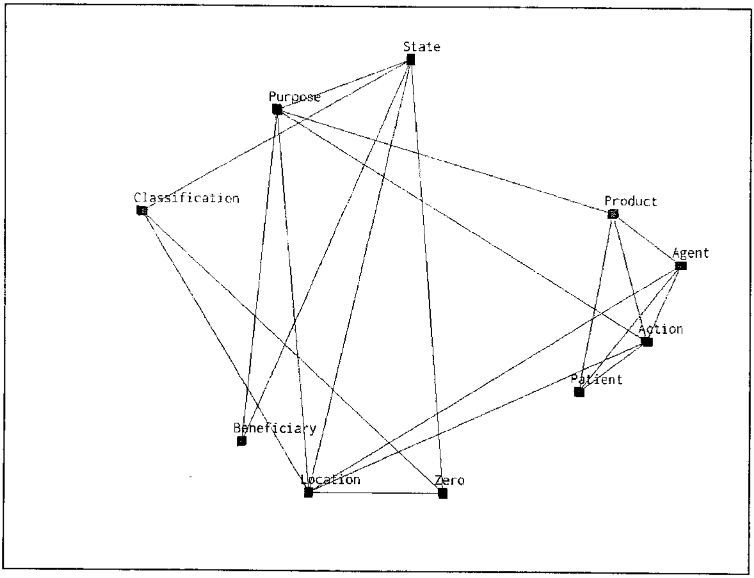

Figure 9. A filtered version of the network graph of semantic roles shown in Figures 7 and 8 showing a network in which the role of purpose appears.

By allowing the user to focus on a selected semantic role, the network that is revealed may be used, as in the 2D implementation above, to confirm or falsify existing definitions and theories, or even to reveal new patterns, suggesting new hypotheses (for example, that the semantic role of purpose does not only occur after controlled predicates, but also in states).

The functionality of the 3D implementation, however, proved to be rather limited. Adding more filters and aggregate functions, as well as manual repositioning and drill-down facilities, would have made the graph much more useful, but this has not been implemented in this version. The 2D version, although visually less fascinating and impressive, is easier to read and interpret and provides more useful text-mining possibilities. This may indicate that, for this type of data-mining activity, a 2D graph may often be preferable, probably because it is easier to read and interpret.

9. CONCLUSION

Visualisation is a graphical display of data that facilitates creative information retrieval and exploration, since it allows researchers to dynamically change the focus of a data-mining process or to implement what-if scenarios. One-, two- and three-dimensional approaches may be used to fulfil various requirements in this regard. XML provides suitable technology to capture or prepare the data that should be visualised, since this mark-up language is very customisable and extensible.

2D and 3D interactive network graphs were, therefore, proposed to enable dynamic, "what-if" investigations, using BH linguistic data of various modules marked up in XML. This concept was applied, as an example, to the networks of semantic roles and frames in Genesis 1:1-2:3.

The 2D implementation proved to facilitate more functionality than the 3D tool because it allows the researcher to drill down to see semantic frame details, as well as the frequencies of the semantic roles in the database. Manual repositioning of the elements on the graph and clearer filtering (in showing unique frames) were more benefits of the 2D graph. Both experiments enable the researcher to move beyond the limits of single clauses and provide an interactive, "playful" way to text-mine the linguistic database. A three-dimensional approach may be more applicable and useful if more than one layer of data has to be visualised simultaneously, but this article looked only at one layer of data (the structural semantics of the text).

Both experiments implemented only a small number of graphical aids. More possibilities that may be researched include the indication of frequencies through the use of colours or by adjusting the weights of the connecting lines, as well as more drill-down facilities, for example, a right-click-activated procedure that shows a pop-up window listing relevant examples. The use of various approaches can be optimised by allowing researchers to switch between various graphical implementations such as a three-dimensional version of the clause cube and two-dimensional slices. Since frames with a low frequency could offer less generally known and therefore more interesting information, it could be helpful for researchers to be able to focus interactively on more interesting frames.

Similar projects by linguists, including Biblical Hebrew scholars, may stimulate new ideas and may be used to formulate new hypotheses, which may then be examined in depth.

BIBLIOGRAPHY

Andersen, F I & Forbes, A D 2003. Hebrew Grammar Visualised: I. Syntax. *ANES* 40, 43-61.

Bradley, J 2003. Finding a Middle Ground between 'Determinism' and 'Aesthetic Indeterminacy': a Model for Text Analysis Tools. *Literary and Linguistic Computing* 18(2), 185-207.

Du Plooy, N F 1998. *An Analysis of the Human Environment for the Adoption and Use of Information Technology.* D.Com. thesis, University of Pretoria.

Eden, B 2005. 3D Visualization Techniques: 2D and 3D Information Visualization Resources, Applications, and Future. *Library Technology Reports* 41(1), 55-64. (Chapter 3: Practical Applications of 2D and 3D Information Visualization for Information Organizations).

Flynn, P 2002. Is there Life beyond the Web? *Literary and Linguistic Computing* 17(1), 49-59.

Freedman, M 2006. Information Visualization Brings Data into Sharper Focus: Analysing Greater Volumes at Improved Resolution that Allows Access for all Levels. *Scientific Computing* 23(3), 24-25.

Huang, M L, Eades, P, & Wang, J 1998. On-line Animated Visualization of Huge Graphs Using a Modified Spring Algorithm. *Journal of Visual Languages and Computing* 9, 623-645.

Keller, T, Gerjcts, P, Scheiter, K & Garsoffky, B 2006. Information Visualizations for Knowledge Acquisition: the Impact of Dimensionality and Color Coding. *Computers in Human Behaviour* 22, 43-65.

Kroeze, J H 2002. Developing a Multi-Level Analysis of Jonah using HTML, in: Cook, J (ed.). 2002. *Bible and Computer: The Stellenbosch AIBI-6 Conference, Proceedings of the Association Internationale Bible et Informatique "From Alpha to Byte" University of Stellenbosch 17-21 July, 2000.* Leiden: Brill, 653-662.

Kroeze, J H 2007. A Computer-Assisted Exploration of the Semantic Role Frameworks in Genesis 1:1-2:3. *JNSL* 33(1), 55-76.

Kroeze, J H 2008. Developing an XML-based, Exploitable Linguistic Database of the Hebrew Text of Gen. 1:1-2:3. PhD(IT) thesis. University of Pretoria. (Unpublished.) Available online (http://upetd.up.ac.za/thesis/available/etd-07282008-121520/).

Kroeze, J H, Bothma, T J D, Matthee, M C & Kroeze, J C W 2008. Visualizing Mappings of Semantic and Syntactic Functions. *Proceedings of the Sixth International Conference on Informatics and Systems (INFOS2008), Cairo, Egypt, 27-29 March 2008.* (On CD.) Available online (http://www.fci-cu.edu.eg/infos2008/infos/MM_10_P061-072.pdf).

Landauer, T K, Laham, D & Derr, M 2004. From Paragraph to Graph: Latent Semantic Analysis for Information Visualisation. *Proceedings of the National Academy of Science of the United States of America* 101, Supplement 1, 5214-5219.

Manning, C D, Jansz, K & Indurkhya, N 2001. Kirrkirr: Software for Browsing and Visual Exploration of a Structured Warlpiri Dictionary. *Literary and Linguistic Computing* 16(2), 135-151.

Neyt, V 2006. Fretful Tags amid the Verbiage: Issues in the Representation of Modern Manuscript Material. *Literary and Linguistic Computing Advance Access,* 1-13.

Petersen, U 1999. *The Extended MdF Model.* [Online]. Available: http://www.hum.aau.dk/~ulrik/pdf/Emdf-A4.ps [Cited 15 October 2004].

Petersen, U 2004. Emdros: a Text Database Engine for Analysed or Annotated Text. *Paper read at the 20th International Conference on Computational Linguistics, Geneva, 2004.* [Online]. Available: http://emdros.org/petersen-emdros-COLING-2004.pdf [Cited 15 October 2004].

Ramsay, S 2003. Toward an Algorithmic Criticism. *Literary and Linguistic Computing,* 18(2), 167-174. (Special Section: Reconceiving Text Analysis.)

Rob, P & Coronel, C 2009. *Database Systems: Design, Implementation, and Management.* 8th ed. Boston, MA: Course Technology.

Sinclair, S. 2003. Computer-assisted Reading: Reconceiving Text Analysis. *Literary and Linguistic Computing,* 18(2), 175-184.

Thuraisingham, B M 2002. *XML Databases and the Semantic Web.* Boca Raton, Florida: CRC Press.

Walsh, N & Muellner, L 1999. *DocBook: the Definitive Guide.* Sebastopol, CA: O'Reilly.

Witt, A 2005. Multiple Hierarchies: New Aspects of an Old Solution, in: Dipper, S, Götze, M & Stede, M (eds) 2005. *Heterogeneity in Focus: Creating and Using Linguistic Databases* (Interdisciplinary Studies on Information Structure 02, 55-85.). Potsdam: Universitätsverlag Potsdam.