WordNet Atlas: a web application for visualizing WordNet as a zoomable map

Matteo Abrate, Clara Bacciu, Andrea Marchetti and Maurizio Tesconi
Istituto di Informatica e Telematica (IIT) - CNR
Via G. Moruzzi, 1 - Pisa, Italy
{matteo.abrate,clara.bacciu,andrea.marchetti,
maurizio.tesconi}@iit.cnr.it

Abstract

The English WordNet is a lexical database containing more than 206,900 word-concept pairs and more than 377,500 semantic and lexical links. Trying to visualize them all at once as a node-link diagram results in a representation which can be too big and complex for the user to grasp, and which cannot be easily processed by current web technologies.

We propose a visualization technique based on the concept of *semantic zooming*, usually found in popular web map applications. This technique makes it feasible to handle large and complex graphs, and display them in an interactive representation. By zooming, it is possible to switch from an overview visualization, quickly and clearly presenting the global characteristics of the structure, to a detailed one, displaying a classic node-link diagram.

WordNet Atlas is a web application that leverages this method to aid the user in the exploration of the WordNet data set, from its global taxonomy structure to the detail of words and synonym sets.

1 Introduction

A big variety of data sources often contain a massive amount of interlinked elements, forming very large graph structures: Telecommunication traffic, social networks and the World Wide Web are all prominent examples of sources that could yield graphs having more than 1 million edges (Abello et al., 2002; Cui, 2011). Because of their size and topology, such structures could be very difficult to explore, even for expert users. Getting an overview of or finding detailed information in them can prove to be complicated, and a poorly designed interface could be of little help if it makes the user feel overwhelmed or disoriented.

The information visualization community agrees that providing a visual representation of graphs is of a great importance for end users. Visual depictions exploit human visual processing to reduce the cognitive load of many tasks that require understanding of global or local structure of data (Munzner, 2000; Ware, 2004; Zhang, 1991).

To obtain such representations, the data must be preprocessed and interpreted. For example, input data can be simplified by highlighting or hiding some features, considered respectively more or less interesting for a given task. The data can also be enriched by derived features.

Size is a major challenge in graph visualization, leading to problems like algorithm complexity, display cluttering, poor readability and difficulties in navigation (Cui, 2011). The aforementioned issues are also found within the context of visualization of smaller-scale graphs, where the size is still a relevant issue (100,000 edges or more). Structures of that size could be obtained from document collections such as Wikipedia, Linked Data[1] datasets like DBpedia or GeoNames, lexical databases like WordNet.

The English WordNet version 3.0 (Miller, 1995; Fellbaum, 1998) contains 155,287 words linked to 117,659 sets of synonyms (synsets) each representing a concept, defining 206,941 word-synset pairs[2]. Synsets are grouped by part-of-speech in four sub-structures: nouns, verbs, adjectives, and adverbs. Moreover, words and synonym sets are interconnected by 285,348 semantic links (semlinks) and 92,231 lexical links (lexlinks) of 28 different types. A complete visual representation of WordNet using a standard node-link diagram would result in a particularly heavy graphic to pro-

[1] http://linkeddata.org
[2] http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html

cess and display, and would probably be too big and complex for users to understand.

Current web technologies cannot cope with such a number of elements without special techniques. These techniques should somehow reduce the amount of information sent to the client and rendered by the user agent, without excessively compromising the meaning of the graphic or worsening the quality of user interaction. Moreover, without choosing a good layout, the WordNet graph could easily render in a large, intricate and unorganized diagram.

This paper describes an interactive map[3] based on a modified radial tree layout. It lets the user navigate the entire structure of the WordNet graph of noun synsets and makes him capable to have a general overview, as well as to concentrate on details on demand (Shneiderman, 1996). We believe that this visualization adds usability and usefulness to WordNet, because it helps the user to get oriented in the large amount of data it contains.

1.1 Outline of the paper

In section 1, we discussed the main issues related to the visualization of graphs obtained from large data sets, such as WordNet.

In section 2, we provide a background reference for general visualization concepts and methods, ending with a brief survey of the state of the art of web interfaces for browsing WordNet (subsection 2.2).

In section 3, we describe the proposed solution, along with a report of the analysis of WordNet's structure (subsection 3.1) and implementation details (subsection 3.2).

In section 4, we present our conclusions and hint at possible directions for further research.

2 Background

2.1 Graph visualization concepts

It is known that our perception of things is mostly due to vision: we acquire more information through sight than through all the other senses combined (Ware, 2004). This is why a good visual representation can be a powerful interface between large amounts of data and our cognitive system, helping us in analysis and decision-making tasks. It has been pointed out by Ware (2004) that a good visualization has five big advantages:

- It gives us the ability to comprehend large amounts of data;
- It lets us perceive properties that where non evident before;
- It makes problems with the data immediately apparent;
- It facilitates us in understanding both large-scale and small-scale features of the data;
- It facilitates hypothesis formation about the data.

To best exploit such advantages, visualizations can be carefully designed in a way that conveys meaning while keeping aesthetics into account. This kind of visualizations are often referred to as *infographics* (information graphics).

In his survey, Cui (2011) describes the main issues behind graph visualization, along with layout techniques, like node-link and space division, and interaction styles, such as filtering and *semantic zooming*. The latter technique improves a typical *zoom and pan* approach by introducing the concept of Level Of Detail (LOD), thus suggesting to display the graph diagram in a *Zooming User Interface*.

Zooming User Interfaces (ZUI) are a kind of graphical interfaces that gives the user the ability to zoom and pan an environment. Within this environment, visual representations of the data are displayed with different Level Of Detail according to the zoom level. Very popular and successful services that embrace this approach are web mapping applications such as Google Maps[4] or Bing Live Maps[5], where the user could quickly zoom out for an overview of the whole planet and zoom in for observing the details of a particular place. It is our opinion that the semantic zooming approach could prove to be valid outside the context of digital cartography, and in particular within the context of visualization of big graphs such as WordNet.

Many tools have been developed to help researchers to visualize graphs, even of big size. These tools can be very useful to analyze a graph to envision a meaningful graphical representation to base an infographic upon.

Gephi[6] is an interactive visualization and exploration platform for various kinds of networks

[3]The web application will be publicly available at http://wafi.iit.cnr.it/wordnetatlas

[4]http://maps.google.com

[5]http://www.bing.com/maps/

[6]http://gephi.org/

and complex systems, dynamic and hierarchical graphs up to 50 thousand nodes and 500 thousand edges. Tulip[7] (Auber, 2003) is a framework to interactively visualize graphs up to 1 million elements, that makes several graph drawing, clustering and visual mapping algorithms available.

Among others, Visone[8] is a research project that designs and implement an interactive visualization software for social network analysis. Cytoscape[9] is an open source platform designed for biological research, now used to display and analyze different kind of networks. NodeXL[10] is a template for Microsoft Excel that integrates a graph visualization and analysis tool into the application.

2.2 Web interfaces to WordNet

There are many different interactive visualizations of the WordNet graph that complement the traditional WordNet Search[11]. They often show words of different part-of-speech, synsets and senses as graph nodes, and word-synset links, semlinks and lexlinks as edges. They show a small subgraph of WordNet, created on the fly. This graph is dynamically adjusted as the user chooses to navigate to other nodes, selected by clicking or by issuing a word search.

WordVis[12], Synonym[13] and Javascript Visual-Wordnet[14] show the graph from the vantage point of a selected word or synset. It centers the selected node in the view, showing details about it and deleting the nodes that become not directly connected to it. WordNet Editor[15], Visuwords[16] and the visualization tool[17] described in the work by Kamps (2002) actually allow a deeper exploration of the graph: the user can click on a node and the connected nodes are added to the view, without deleting the old ones. The disadvantage of these approaches is the progressive performance deterioration as nodes and links are added. Treebolic[18] uses a different kind of graph visualization compared with the previous ones: the nodes position is computed as soon as the graph is drawn, they remain still (they don't adjust their position to better fit the screen), and a *fisheye view* (Furnas, 1986) is used to focus on the details. Even in this case, though, the graph is partial and centered in a specific word sense. Collins (2007) uses a fisheye technique too, giving detail as well as a good level of overview. The showed graph is never complete, though, and it is always rooted in the selected node.

3 WordNet Atlas

The potentially huge amount of nodes and edges of a graph could be organized in a meaningful spatial layout, alongside artificially added spatial features such as colored regions or placemarks, possibly derived from graph data or measures. The resulting visualization will be a mix between a classic node-link diagram and a so-called infographic, a graphic way to present information that abstracts and represents many qualitative and quantitative aspects of a subject in a carefully studied design.

To make the visualization and exploration of the WordNet graph more convenient for the users, we tried to make it easier for them to get an orientation and remember where the nodes are in the map, and consequently to enhance their ability to find them again (revisitation). This is achieved by drawing an interactive graphic that gives a stable representation of the synset nodes (they are always drawn in the same position) and offers the users an overview of the entire data set, along with some added *landmarks*, useful to improve their own orientation. Since the diagram is intended for presentation and for a better fruition mostly by non experienced WordNet users, some simplification in data has been made. Therefore, the map is far from being a perfect representation of the English WordNet lexical database.

In WordNet Atlas, we choose to show two different visualizations at once, one best suited for exploring the graph of synsets, and one for navigating among words (see figure 1).

The first is an interactive map of a graph, representing all the WordNet noun synsets arranged in a special circular layout. At the center of the map is the most generic node of the noun synset taxonomy, the one containing the word "entity"[19], serving as the root of a radial tree layout.

To further characterize the representation, we

[7]http://tulip.labri.fr/TulipDrupal/
[8]http://www.visone.info/
[9]http://www.cytoscape.org/
[10]http://nodexl.codeplex.com/
[11]http://wordnetweb.princeton.edu/perl/webwn
[12]http://wordvis.com/
[13]http://code.google.com/p/synonym/
[14]http://kylescholz.com/projects/wordnet/
[15]http://wordventure.eti.pg.gda.pl/wne.html
[16]http://www.visuwords.com/
[17]http://staff.science.uva.nl/ kamps/wordnet/
[18]http://id.asianwordnet.org/visualize/treebolic/
[19]http://wordnet.princeton.edu

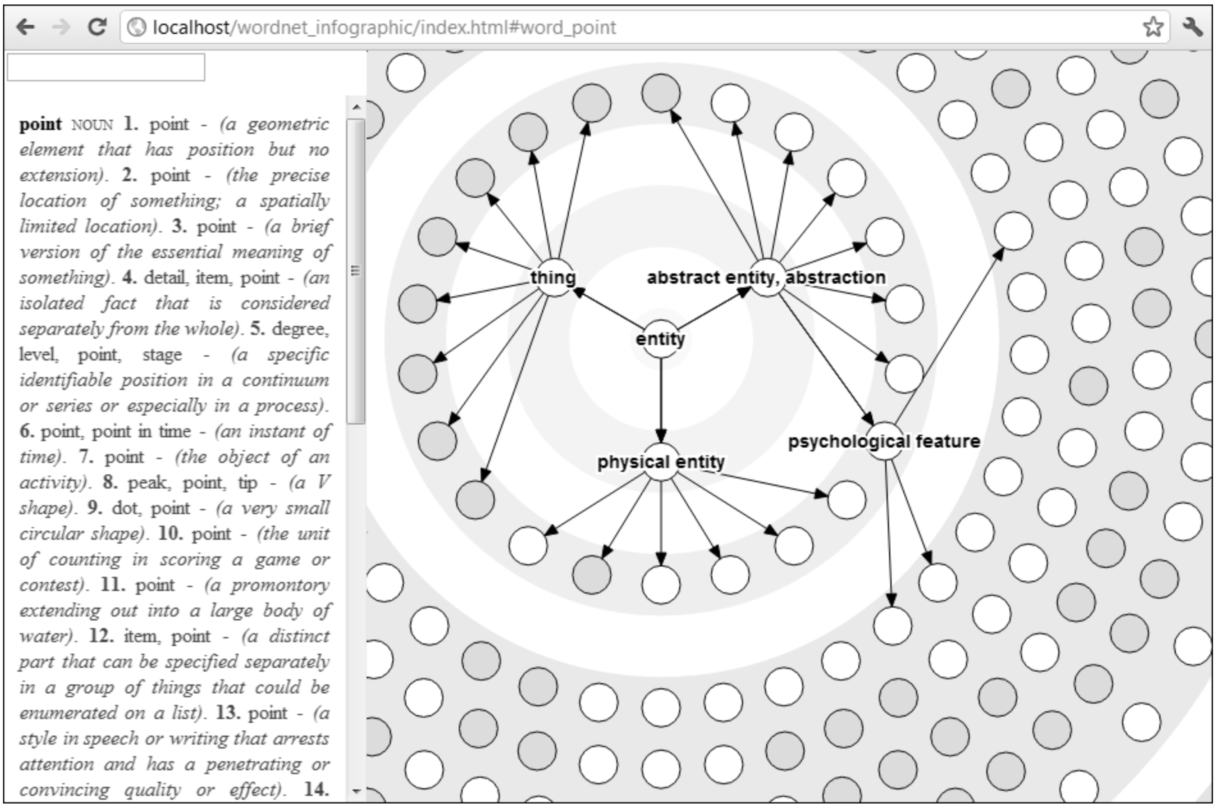

Figure 1: Screenshot of WordNet Atlas prototype interface.

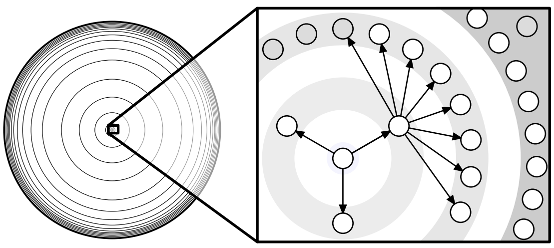

Figure 2: Different Levels Of Detail: at minimum zoom (left), the whole graphic of the noun synsets, divided into rings, and at a greater zoom (right), the central region, in which the classic node-link diagram is visibile.

decided to display each level of the tree as a colored ring underlying the representation of nodes. This should also have the effect of enhancing the graph memorability by leveraging the user's spatial memory (Ghani and Elmqvist, 2011).

To address both the user's inability to cope with large amounts of displayed symbols and the technical problems of performance of current web technologies, we decided to follow a *semantic zooming* approach.

At the minimum Level Of Detail, the entire graph is shown as a circle made of concentric rings, representing the depth level of the taxonomy tree. The ring thickness depends on how many synset are present at that depth level. Rings play the role of *spatial landmarks* (Ghani and Elmqvist, 2011) while giving an overview of the taxonomy tree structure.

By zooming in, the user can load the actual synset nodes grouped in the rings, shown as white circles. Rings are still displayed as colored regions under the graph, following the technique of *substrate encoding* (Ghani and Elmqvist, 2011). The user can use this visual information to more easily recognize the location of the map he or she is looking at.

By interacting with the synset, the user could reach the maximum level of detail, obtaining fine grained information such as the synset's definition, its set of synonym words and semantic links (edges) that connect it with other synsets in different points of the map.

We developed a second visualization, complementary to the map: an hypertextual dictionary shown at the left side of the interface. The user can search for a word entry by using the provided search box, or scroll the whole dictionary by dragging a scroll bar. Like a paper dictionary, each entry displays a word, its part-of-speech and a list of numbered word senses. For each sense, a list of synonym words and a definition are displayed. Not unlike the map, we decided to give the user an overview of the amount of available information in the word database by giving them the illusion that the whole dictionary was loaded in a scrollable section of the interface.

It is possible to jump from a word to another within the dictionary by clicking it in the synonym lists or in the map, or to jump to a synset within the map by clicking its definition. This behavior integrates seamlessly with the standard user agent interface, so the user is able to exploit the familiar navigation buttons, the browser history and bookmarks to navigate words and synsets.

3.1 Analysis of the WordNet database

In order to find useful properties to obtain the described visualization, we analyzed an SQL version of the WordNet 3.0 database.

We chose the synsets nouns as graph nodes, trying to identify a hierarchy rooted in the node "entity", and hypernymy-hyponymy ("is-a") links as edges, since they are the most frequently encoded[20] and more likely to form a meaningful structure, resembling a taxonomy. Even with this simplification, the remaining nodes and edges do not define a simple tree, but define instead a *tangled tree*, i. e. a tree that contains a few nodes that can have multiple parents.

Par.	Nodes	Internal	Leaves	Types	Inst.
0	1	1	0	1	0
1	79,901	16,680	63,221	72,962	6,939
2	2,134	463	1,671	1,388	746
3	63	12	51	30	33
4	12	0	12	3	9
5	3	1	2	1	2
6	1	0	1	0	1
Total	**82,115**	17,157	64,958	74,385	7,730

Table 1: Count of synset nodes, grouped by number of parents.

We decided to consider the root synset 100001740 (entity, "That which is perceived or known or inferred to have its own distinct existence (living or nonliving)") as the map center, and to place the remaining nodes around it, using a modified radial tree layout. For each synset, we computed the shortest path to reach the root node through hyponym links[21] as a measure of tree depth. Then, we grouped the nodes in rings by depth value. Unlike a standard tree layout, we decided to place the nodes nearer to each other, by packing the nodes of a ring in more than one orbit. By using a traditional radial layout, in which all the nodes of a ring are placed in a single orbit, rings would have to be drawn very far from each other, because of the fast growth in the number of nodes as the ring index increases.

Ring	Nodes	Internal	Leaves	Types	Inst.
0	1	1	0	1	0
1	3	3	0	3	0
2	22	12	10	22	0
3	228	146	82	227	1
4	2,020	891	1,129	2,011	9
5	6,249	2,033	4,216	5,641	608
6	12,267	3,171	9,096	10,555	1,712
7	18,936	3,276	15,660	16,892	2,044
8	14,155	2,666	11,489	13,028	1,127
9	11,042	1,955	9,087	9,286	1,756
10	7,207	1,251	5,956	6,867	340
11	4,267	738	3,529	4,204	63
12	2,505	451	2,054	2,450	55
13	1,383	261	1,122	1,381	2
14	846	135	711	845	1
15	449	99	350	448	1
16	341	57	284	341	0
17	164	11	153	153	11
18	30	0	30	30	0
Total	**82,115**	17,157	64,958	74,385	7,730

Table 2: Count of synset nodes, grouped by ring. The ring index corresponds to the minimum depth of the node (length of the shortest path to the root node).

WordNet synsets are divided in two categories: types and instances. We identified and marked 7,730 instance noun synsets (target of instance hyponym links[22]) to differentiate them visually in the map. We also marked the leaf nodes, 64,958 out of the total of 82,115 noun synsets (79%).

3.2 Implementation details

Displaying a graph with more than 80,000 nodes is a daunting task, even for state-of-the-art web technologies. Moreover, a visualization of a graph that big would probably be unsuitable for giving the user an overview of it. We decided to compute offline the fixed position of each synset node representation within the space of the graphic, and to

[20] http://wordnet.princeton.edu/

[21] Both regular and instance hyponym links, though in the visualization they are differentiated.

[22] Type nodes are always destinations for regular hyponymy edges only, and instance nodes are destinations for instance hyponymy edges. This is true except for 5 specific nodes that are probably errors in the database.

store the resulting data in a geospatial database[23]. This kind of databases provide spatial indices, to achieve good performance when issuing spatial queries. For example, a spatial query could request all the countries that overlap a specified polygon in a map. We implemented a REST-style (Fielding, 2000) web API that makes use of this technique to retrieve a JSON[24] representation of the nodes that overlap the rectangular region defined by the client viewport. This approach ensures that only the visible nodes are actually loaded from the server and displayed by the user agent, while giving to the user the illusion of having the whole graph loaded when panning.

When the user decreases the zoom, the viewport increases the covered region, and consequently the amount of information to load and show. As previously described, we followed a common approach used for Zooming User Interfaces, and reduced this amount of data by implementing a concept of Level Of Detail: each graphic element stored in the database is given a minimum zoom level for it to appear. The spatial database receives the current zoom setting of the client along with the position and size of the viewport, and discards graphic symbols that are considered too fine-grained to appear at that particular zoom level.

The JSON response returned by the web API is then processed client-side and transformed into an interactive vector graphic by a cross-browser[25] JavaScript library[26] that leverages HTML5 canvas[27] and Scalable Vector Graphics (SVG)[28] technologies.

The hypertextual dictionary view of WordNet is implemented by using similar techniques, applied to the single scroll dimension instead of the 2D space of the map.

The selection of words and synsets is implemented by following a common web application practice that requires to update the page hash URI whenever a meaningful change in application state is detected. In our work, this makes it possible to use the browser's history, navigation buttons and bookmarks to browse words and synsets, each uniquely identified by an URI. This enables the user to perform the navigation task effectively, by leveraging proven user interaction paradigms and familiar interfaces. Each item presented by WordNet Atlas is thus uniquely identified by a dereferenceable URI[29] that can be shared or simply written in a document to unambiguously refer to the item.

4 Conclusions

We presented WordNet Atlas, a web application capable of giving WordNet users an interactive visualization of the resource. We believe that its capability to scale from a meaningful overview to the details of WordNet structure makes it ideal for presentation or teaching purposes, or for the promotion of WordNet to a broader audience. We also believe that the effort to investigate a graphical way to organize such a big graph could assist in the inspection and improvement of the underlying database, for example by exposing anomalies in the structure caused by human errors.

4.1 Future work

In the next step of our work, we will search for confirmation of the usefulness of the presented visualization technique. We are already setting up a round of user evaluation tests, with comparisons with interactive representations of WordNet found in literature.

In our future research, we hope to improve both the quality of the interaction and the amount of information the application is capable of conveying, while keeping its design polished. In respect to that, the visualization layout can probably be improved by taking into account more information, such as the node height (length of the path to a leaf) or the distinction between type and instance nodes. We plan to provide a visualization for the other three WordNet subnets, verbs, adjectives and adverbs, linked to the current one.

It could be interesting to use the same system behind WordNet Atlas to display wordnets in other languages, and to compare and interlink the resulting representations.

The described methodology could also prove to be useful to represent other big graphs, such as social graphs or semantic web graphs, and even data sets with a non-graph structure.

[23] http://postgis.refractions.net

[24] http://www.json.org

[25] WordNet Atlas is reported to work in Mozilla Firefox 4+, Internet Explorer 8+ and Chrome 14+.

[26] http://raphaeljs.com

[27] http://www.w3.org/TR/html5/the-canvas-element.html

[28] http://www.w3.org/Graphics/SVG/

[29] This is a best practice in web development, commonly used in the semantic web initiative (http://semanticweb.org).

References

James Abello, Jeffrey Korn, and Matthias Kreuseler. 2002. Navigating giga-graphs. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '02, pages 290–299, New York, NY, USA. ACM.

David Auber. 2003. Tulip: A huge graph visualization framework. *Graph Drawing Softwares*, pages 105–4126.

Christopher Collins. 2007. Wordnet explorer: Applying visualization principles to lexical semantics.

Weiwei Cui. 2011. A survey on graph visualization.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, MA.

Roy Thomas Fielding. 2000. Architectural styles and the design of network-based software architectures.

George W. Furnas. 1986. Generalized fisheye views. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 16–23.

Sohaib Ghani and Niklas Elmqvist. 2011. Improving revisitation in graphs through static spatial features. In *Proceedings of Graphics Interface 2011*, pages 175–182.

Jaap Kamps. 2002. Visualizing wordnet structure. In *Proc. of the 1st International Conference on Global WordNet*, pages 182–186.

George A. Miller. 1995. Wordnet: a lexical database for english. *Commun. ACM*, 38:39–41, November.

Tamara Munzner. 2000. Interactive visualization of large graphs and networks.

Ben Shneiderman. 1996. The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Symposium on Visual Languages*, pages 336–343.

Colin Ware. 2004. *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Jiajie Zhang. 1991. The interaction of internal and external representations in a problem solving task. In *Proceedings of the Thirteenth Annual Conference of Cognitive Science Society*, pages 88–91.