"Andre ord" – a Wordnet Browser for the Danish Wordnet, DanNet

Anders Johannsen
University of Copenhagen
Copenhagen, Denmark
ajohannsen@hum.ku.dk

Bolette S. Pedersen
University of Copenhagen
Copenhagen, Denmark
bspedersen@hum.ku.dk

Abstract

A publically available wordnet browser will, if it does not remain in obscurity, have to cater to two different audiences: the professional lexicographers and the general public. This demonstration paper describes the wordnet browser "Andre ord" which has been developed for the Danish wordnet, DanNet. The first version was released in autumn 2009, followed by the release of a refined version in late 2010. The browser applies the open source framework Ruby on Rails and the graphing toolkit Protovis, and is itself open source. In the paper we discuss what design compromises might be needed when accommodating professionals and non-specialists alike, although our main concern is giving the general public an intuitive impression of the resource. To this aim we adopt the familiar, dictionary-like word-in-synset as the basic unit of the browser idea, but at the same time try to convey the idea that every piece of information in the wordnet is located somewhere in a larger semantic network structure.

1 Introduction and related work

In a certain sense, wordnet browsers abound. No doubt owing to its generous open source licence, content from the Princeton WordNet crop up in all sorts of dictionary-like applications. We have encountered it such diverse places as iPhone applications, writing assistants, web-based dictionaries, and as stand-alone software. In most of these settings the wordnet structure is disregarded, and the content is couched into an ordinary dictionary format that is familiar to most people. For our present purposes we will not consider these as wordnet browsers, but direct our attention to three of the comparably fewer that offer a full experience.

The official Princeton Wordnet browser[1] (Princeton University) exposes a text-like interface to the wordnet. A query for a word brings up a page that lists the various synsets it is a part of. Each synset can then be expanded by a click, which reveal its relations to other synsets along with additional facts about the synset, such as its allowed sentence frames. Hierarchical relations, which occur in for example "inherited_hypernym", are marked by indentation. All of the data in Wordnet is accessible by this interface and no attempt is made to restrict the output, even when this leads to pages of unwieldy length.

Visuwords[2] conceptualizes Wordnet as a network structure and employs a force-directed graph layout to visualize the connections between synsets. This type of layout algorithm, which progresses by simulating the graph as a physical system, with edges behaving like, for instance, springs, and nodes as charged particles, takes a short while to settle down, making the graph appear very lively at first. When entering a word it constructs a graph centered on a node with that label, linking to all synsets that include the query word. Ambiguity is thus not resolved but purposely kept in the graph illustration. Wordnet relations that originate in the synsets are also drawn, although if there are more than a few of each type they seem to get capped arbitrarily. Visuwords use color coding extensively, both on its own, to distinguish between synsets in various parts-of-speech, and in combination with shapes for telling different relation types apart.

While not a full-fledged wordnet browser, Nodebox[3] does contain an inspirational visualization of a wordnet synset. It uses a radial node-link diagram, which packs nodes along the radius of a circle. Edges are drawn by the use of line segments. Contrasting with Visuwords, the layout is static, and less dense with just one disambiguated synset and a single relation type being

[1] http://wordnetweb.princeton.edu/perl/webwn

[2] http://www.visuwords.com/

[3] http://nodebox.net/code/index.php/WordNet

Bolette Sandford Pedersen, Gunta Nešpore and Inguna Skadiņa (Eds.)
NODALIDA 2011 Conference Proceedings, pp. 295–298

displayed at a time, improving the over-all readability as well as calming the appearance of the graph. Because only one kind of information is visualized the graph does not use any color or shape encoding.

The rest of the paper is organised as follows. We introduce the DanNet resource in Section 2, and Section 3 describes the design considerations made. Technical details of "Andre ord" are given in Section 4. Finally, Section 5 briefly presents some ideas for future implementation.

2 Presentation of the resource: DanNet

DanNet (cf. wordnet.dk) is an open-source, lexical-semantic resource for Danish built in collaboration between The University of Copenhagen and Det Danske Sprog- og Litteraturselskab. The resource is meant for integration in computational systems that include a semantic aspect, such as writing aids and intelligent information navigation systems. Currently, it has been integrated in the Danish version of OpenOffice where it is used as a facility to suggest broader and more narrow terms, and it is integrated in a search module developed for The Municipality of Odense by the Danish company LAT-computing. Furthermore, the resource has been used in several research projects concerned with word sense disambiguation and search.

DanNet is a classical wordnet in the sense that it conforms to the framework of Princeton WordNet (Fellbaum 1998) and EuroWordNet (Vossen (ed.) 1999) with a few exceptions. However, in contrast with most other wordnets, DanNet has been constructed using a merge approach where the wordnet is constructed monolingually (based on Den Danske Ordbog) and thereafter linked to Princeton WordNet. This strategy can be seen in contrast to the more widely adopted expand approach where synsets are translated from Princeton WordNet into the target language.

At the time of writing, DanNet contains 62,000 synsets and is still under development within the DK-CLARIN project, until mid of 2011. DK-CLARIN is the Danish branch of the EU project CLARIN, an acronym which expands to a common language resources and technology infrastructure.

3 Design considerations

A publically available wordnet browser will, if it does not remain in obscurity, have to cater to two different audiences: the professional lexicographers and the general public. Sadly, their expectations and skill sets do not always align. To most people a wordnet will not be something readily familiar, and so the concept of, for instance, a synset will have to be set down before an uninitiated user can make sense of relations between such entities. This is a major challenge since the famously impatient web surfers of today do not like prolonged explanations.

However, as nearly everyone knows their way around a standard dictionary where a headword leads to a definition, emphasizing the similarity between a wordnet and a dictionary, rather than pointing out the differences, might reduce the burden of explanation. This has lead us to adopt the word-in-synset as the basic unit of the browser, that is: a synset pinned down by a particular choice of one of its synonyms. Each word-in-synset is presented on a separate page. Even though such a page in effect shows a synset, the notion itself is never brought to attention of the user. For browsing purposes this eliminates the need to explain the more abstract concept of a synset, but still preserving the relational nature of the wordnet.

Furthermore, we have sought to enhance the ease of use of the browser by shifting the display of important relation information from what would, in some cases, require some very long tables to a single, prominently placed graph. Our particular choice of graph, which is accounted for in the next section, very compactly encodes hundreds of relations. Even so there sometimes is a conflict between completeness and comprehensiveness. If the number of relations exceeds a certain limit they can no longer be displayed in the graph without sacrificing readability. In that case we favor comprehensiveness and drop relations according to a developed scheme. Luckily, we only have to resort to this option in 0,2 % of the cases.

In the wordnet browsers surveyed in the "Related works" section of the paper, pages are generated on the basis of a word, a possibly ambiguous string entered by the user and corresponding to one or more synsets. Thus each page view is often required to serve information about multiple synsets that are unrelated (in the wordnet sense), and perhaps distributed across different parts-of-speech. As such this agglomeration of synsets does not provide insight into the structure of the wordnet. We avoid bewildering the user by having him go through a disambiguation process in case of ambiguity at the end of which a single word-in-synset is chosen for display.

Another important consideration is how to highlight the situatedness of the data. We wanted to convey to the user the idea that every piece of information in the wordnet is located somewhere in a larger semantic network structure. We found no single way of effectively communicating this, but rely instead on the combination of several cues; the shortest path to the top node, which is printed like a breadcrumb, the relations to adjacent synsets being displayed in a manner suggestive of a network structure, and a chart that summarizes the complete hierarchy of hyperonomy relations that terminate in the current synset.

Returning, finally, to the problem of the dual audiences: the professionals who arrive with certain theoretically founded expectations, and the casual visitors with a more fleeting interesting in linguistics. We had the good fortune of not being in charge of developing the only browser for the DanNet project, albeit the only public one, and that allowed us to maintain a focus on the laymen perspective since a custom in-house tool already existed. Our concern with the professional audience was consequently less to indulge their desires for specialist functionality and more to make sure that our depiction of the resource was still valid and sound according to their point of view. For while the resource should be as easily accessible as possible, it should be no more so, to quote a phrase; under no circumstances did we want to distort the content of the wordnet to make it easier to understand.

So even if “Andre ord” was not designed with lexicographers in mind, they have nonetheless derived much utility from it. Perhaps owing to the more visual nature of this browser, it has proven very effective at spotting, for instance, relation type and inheritance errors.

4 Technical description

“Andre ord” is a web application, deployed at http://andreord.dk. It is built using the open source framework Ruby on Rails[4], and is itself open source. Protovis[5] from the Stanford Visualization Group is the foundation on which the graphs are constructed. Here we provide an overview of the elements of the central word-in-synset page as well as what steps are needed to arrive at that page.

Before any data from the wordnet can be displayed the user must type in a query. The query should be a single, uninflected word that exists in the wordnet. Helpful suggestions from the database continuously guide the match.

If the query corresponds to a single word-in-synset, it is displayed. Otherwise the user is redirected to a disambiguation page. Here, a list of matches is displayed along with their glosses. Furthermore, each word-in-synset is assigned a unique heading to make them easier to distinguish from each other. The heading is typically the word itself joined by either a hyperonym, a hyponym, or, in case they are not unique, a counter. A partial listing for the word “dronning” (queen) is “dronning (insekt)”, “dronning (dame)”, “dronning (kort)”, and “dronning (regent)” (the translations for the parenthesized words are: “insect”, “lady”, “playing cards”, and “ruler”). These (particular) headings are obviously very helpful for disambiguation.

On the main page two types of visualizations can be toggled. The first one, depicted below, is preselected (“blæseinstrument” is wind instrument).

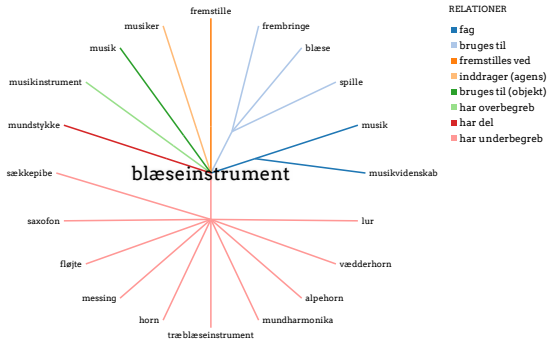

Figure 1: Word-in-synset relations

Here all relations that have the currently displayed synset as a source is shown. The graph is modelled as a two level node-link diagram. Relations connect via an intermediate relation type node, visually clustering nodes that share relation type. Color is used to mark the relation type and can be resolved in the accompanying legend. Graph layout is remarkably simple. Since the number of edge nodes is known in advance, and the level count is fixed, it suffices to group the relations by relation type, plot them as equally spaced points on a circle with a chosen radius, then connect them by line segments to a point on a smaller, inner circle which sits exactly midway between the extremes of the relation type group.

[4] http://rubyonrails.org/
[5] http://vis.stanford.edu/protovis/

As a final step the points on the inner circle are connected to the centre.

The graph can be navigated by clicking. Mouse-over reveals further information about the word-in-synset below the pointer, such as gloss and associated words.

If the number of relations on a given synset exceeds the capacity of this visualization, a capping policy kicks in. It works by grouping the relations by type, then iteratively picking one relation for display from each group, stopping when the limit has been reached. The idea is that smaller relation groups are shown in their entirety while larger ones get allotted the same share, regardless of their size. Fortunately, capping is rarely necessary as the radial layout, coupled with slanted labels, can fit a rather large number of relations.

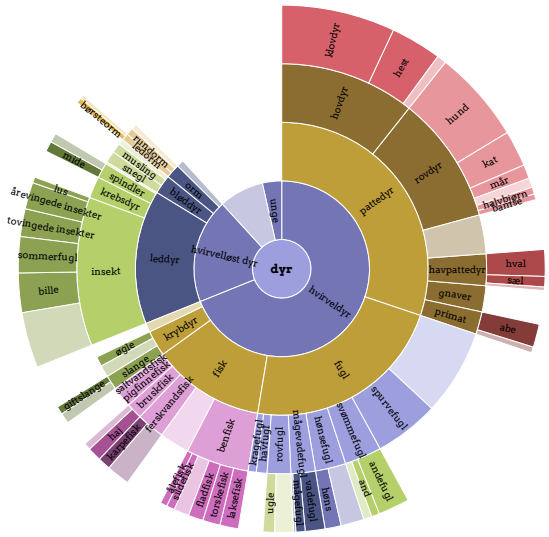

Figure 2: Aggregated hyponyms for "dyr"

The other visualization explores in depth the hyponym relation. In non-technical terms, it answers the question: what is this concept composed of? The answer comes in the form of a sunburst diagram, which is especially well suited for expressing these sorts of nested composed-of relationships. An important thing to notice is that proportionality is preserved so it is possible to visually compare the size (recursively enumerated hyponym count) of each hyponym.

There are synsets, typically situated near the top of the concept hierarchy, for which the number of recursive hyponyms far exceed what can be shown in the diagram. We decide on the most important hyponyms in the following manner. First, the recursive traversal is limited to a depth of four levels, and edge synsets are disregarded. Second, an algorithm is run that collapses synsets too small to be shown (measured by the angular space available for the label) into a special "rest" category, marked by the use of a translucent color and no label. The reason for keeping synsets in the rest category is to maintain proportionality.

Apart from the visualizations the page contains pragmatic language use examples, a trace of the shortest path to the top synset, a complete list of hyponyms with glosses, and lastly, the ontological type of the synset.

5 Conclusion and future work

The development and release of "Andre ord" has highly increased the interest and knowledge of DanNet in the Danish community, both among linguistic and computational researchers and among commercial developers. We are delighted, and also a bit surprised, of the volume of the feedback. It shows that there is value in designing something not meant for ourselves.

Regarding future extensions, the following issues are under consideration. Currently there is no way to distinguish between an inherited relation (typically shared between sister terms) and a relation defined directly on the synset, even though the latter is considered more important. This extra dimension could be encoded using, for example, shape, such as a dotted line, or by attaching a slight glow to the line representing the relation. Another concern is how to heighten the layman user's appreciation of the subtler details of the wordnet. Though it is our firmly held belief that a public wordnet browser should be intuitive and largely self-explanatory, a little linguistic background information here and there may not be that harmful to the enterprise.

References

Fellbaum, C. (ed). 1998. *WordNet – An Electronic Lexical Database*. The MIT Press, Cambridge, Massachusetts, London, England.

Pedersen, B.S, S. Nimb, J. Asmussen, N. Sørensen, L. Trap-Jensen, H. Lorentzen. 2009. DanNet – the challenge of compiling a WordNet for Danish by reusing a monolingual dictionary. *Language Resources and Evaluation, Computational Linguistics Series*. Volume 43, Issue 3:269-299.

Princeton University "About WordNet." WordNet. Princeton University. 2010. <http://wordnet.princeton.edu>

Vossen, P. (ed). 1999. *EuroWordNet, A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, The Netherlands.

ISSN 1736-6305 Vol. 11
http://hdl.handle.net/10062/16955