Data Structures in Lexicography: from Trees to Graphs

Michal Měchura

Natural Language Processing Centre
Faculty of Informatics, Masaryk University
Botanická 68a, 602 00 Brno, Czech Republic
michmech@mail.muni.cz

Abstract. In lexicography, a dictionary entry is typically encoded in XML as a tree: a hierarchical data structure of parent-child relations where every element has at most one parent. This choice of data structure makes some aspects of the lexicographer's work unnecessarily difficult, from deciding where to place multi-word items to reversing an entire bilingual dictionary. This paper proposes that these and other notorious areas of difficulty can be made easier by remodelling dictionaries as graphs rather than trees. However, unlike other authors who have proposed a radical departure from tree structures and whose proposals have remained largely unimplemented, this paper proposes a conservative compromise in which existing tree structures become augmented with specific types of inter-entry relations designed to solve specific problems.

Key words: e-lexicography, dictionary writing systems, placement of multi-word items in dictionaries, bilingual dictionary reversal

1 A brief history of computerization in lexicography[1]

Following Atkins and Rundell [2, p. 3], there are three stages in the dictionary-writing process where computer software comes in: (1) as **corpus query systems** for discovering lexical knowledge in corpora, (2) as **dictionary writing systems** where lexical knowledge is encoded into a form suitable for presentation to human readers and (3) as **websites, apps** etc. which deliver the dictionary onto a user's screen. Together these three areas constitute the discipline known as *e-lexicography* (a good introduction to which is [5]).

[1] It is important to clarify that, in this paper, the term *lexicography* means writing dictionaries for humans: a discipline whose goal is not only to discover the properties of words (a goal it shares with lexicology) but also to communicate those discoveries successfully to human consumers who are neither lexicologists nor lexicographers: to "identify the most effective ways to present the linguistic properties of words in dictionaries according to specific criteria such as the type of dictionary, the intended user group, etc." [7, p. 4]. This separates human-oriented lexicography from computational lexicons such as WordNet [4].

Aleš Horák, Pavel Rychlý, Adam Rambousek (Eds.): Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2016, pp. 97–104, 2016. © Tribun EU 2016

ost innovation in e-lexicography has happened in (1) corpus query systems: so much, in fact, that corpus-driven methods have redefined dictionaries from intuition-based prescriptions to evidence-based descriptions. At the other end of the pipeline, in (3) dictionary publishing, websites and other electronic media had for long only imitated the behaviour of paper dictionaries. Lately, however, some innovation started appearing in this area as new methods of delivering dictionary content to users are emerging while dictionaries are becoming divorced from the original print medium, see e.g. [11].

The area where the least amount of innovation has happened until now is the middle part, (2) dictionary writing. Even though dictionary writing has become completely computerized in the last few decades, the structure of dictionaries we write today has not changed since pre-computer times. Yes, today's dictionary entries tend to be more easily navigable due to generous use of colour, font and whitespace, but that is only a superficial difference in formatting. Yes, today's dictionary writing software ensures that dictionary entries comply with a given schema, but this only replicates what lexicographers would be doing on paper or in a word processor anyway, only with more effort and less perfection. The underlying paradigm has not changed: a dictionary entry is still the same tree structure in which elements such as headwords, senses, part-of-speech labels and example sentences are stacked inside each other by means of parent-child relations where each child has at most one parent. The fact that we still model dictionary entries as trees means that some aspects of the lexicographer's work remain unnecessarily difficult.

2 What we can't do with dictionaries

Here we introduce two well-known problems in lexicography, each of which can be understood as an inconvenient consequence of the tree-like data structure dictionaries are encoded in.

2.1 Placement of multi-word items

Deciding under which headword a multi-word phraseme should be placed is a classical problem in lexicography [3]. Should an item like *third time lucky* be included under *third*, *time* or *lucky*? Arguably the best answer is 'all of them' but the only way to make it appear under all relevant headwords is by copying it. The traditional data structure of dictionary entries as trees imposes the inconvenient constraint that information cannot be shared across multiple entries (other than by copying). This difficulty can of course be worked around further downstream by clever search algorithms, by some form of indexing or cross-referencing, but it would be smarter to fix the problem at source by devising a data structure that allows fragments of entries to be 'shareable', able to appear in multiple entries. This is impossible in a tree structure where each phraseological element can have only one parent, but it is perfectly possible in a graph structure where it can have multiple parents, giving us a method to model many-to-many relations between entries and phrasemes.

2.2 Bilingual dictionary reversal

Another well-known problem in lexicography is reversing a bilingual dictionary [10]. Once we have written a bilingual dictionary from language X to language Y, it is far from trivial to convert it into a dictionary that goes in the opposite direction, from language Y to language X. There are points of indeterminacy which prevent us from doing it completely automatically. More importantly, the process is a one-way street: once we have reversed the dictionary, we have lost the connection between the source and the target: each entry in each dictionary is its own tree structure with no explicit links between them. If and when the source dictionary changes, the reversed dictionary has potentially become outdated as there is no automated way to project changes from one into the other. A more attractive proposition would be to encode pairs of bilingual dictionaries in a structure that keeps them synchronized, so that every element in every entry in the reversed dictionary 'knows' which element in which entry in the original dictionary it came from, and can react to changes. Again, this calls for a graph-based data structure where each element can have relations with other things besides its hierarchical parent.

3 Are graphs the answer?

While trees are the conventional data structure in human-oriented lexicography, lexicons for machines are often encoded as graphs. A typical example is WordNet [4] and other semantic networks which, in effect, are models of the mental lexicon. These seem like a promising source of inspiration. Instead of writing a tree-structured dictionary, one could build a graph-based model of the mental lexicon and then **derive** dictionaries from it, automatically and on demand. The conventional tree-structured entry would become a non-persistent output format, one of many possible 'views' of the graph, while problems such as multi-word item placement and dictionary reversal would disappear. In practice, however, all attempts to build a human-oriented dictionary in this way have so far remained experimental (e.g. [12]). It seems that the lexicography industry is not (yet?) prepared to 'think outside the tree' – or is perhaps the idea itself unrealistic because the lexical needs of humans and machines are incompatible?

Lately, some dictionary publishers have become inspired by the Semantic Web and started experimenting with re-encoding dictionaries as RDF graphs (e.g. [1], [8]). This is a more realistic attempt at innovation because, unlike semantic networks *à la* WordNet, it does not attempt to model the mental lexicon. Instead, it merely captures the same information dictionaries already have in trees and encodes it in a graph. In an RDF graph, dictionary entries can be augmented with various relations which 'break out' of the tree paradigm, for example sense-to-sense links between synonyms. The relations envisaged above, such as many-to-many relations between multi-word phrasemes and word senses, could be accommodated in an RDF graph easily. However, the disadvantage of RDF graphs (and graphs in general) is that they are not as easily human-readable as XML trees (and trees in general), not to mention human-writeable. Trees can be visualized neatly as two-dimensional objects, while graphs often can't. Trees are easy for humans to grasp mentally, while graphs are more difficult to 'take in'. For this reason, it is unlikely that lexicographers will switch to authoring graph-based dictionaries directly any time soon. All RDF encodings of human-oriented dictionaries have so far been automatic conversions from pre-existing tree-structured XML.

The problem then is that, while graphs are the more adequate structure for dictionaries, trees are more 'lexicographer-friendly'. What we need is a compromise: a set up which keeps dictionaries in a tree-like structure as much as possible, but which also allows them to 'break out' of the tree when necessary: for example to allow the sharing of phraseological subentries between entries. Importantly, we also need a dictionary writing system which allows lexicographers to work with dictionary entries in the familiar tree format as much as possible, while only forcing them to 'think outside the tree' when necessary.

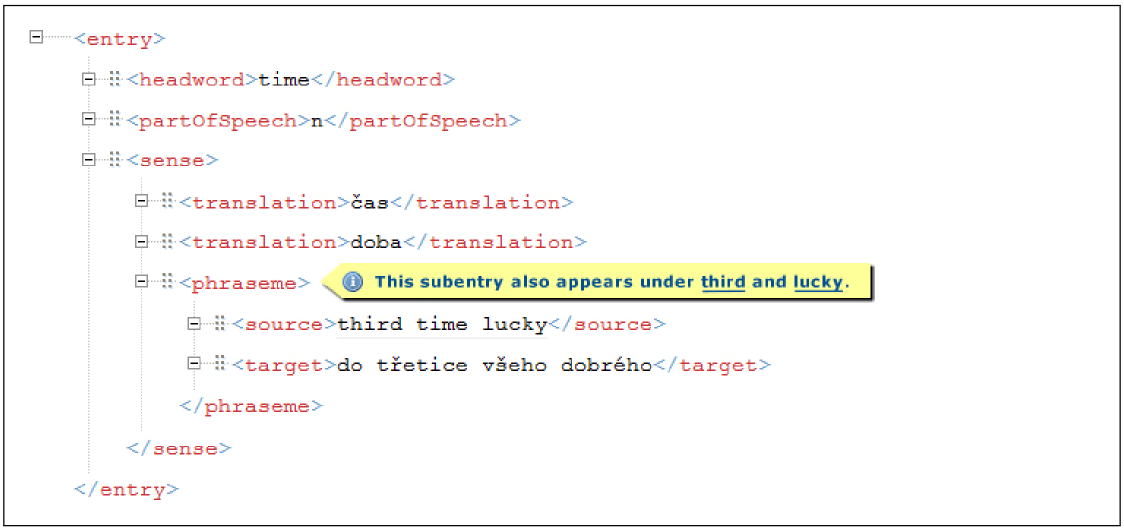
4 Introducing graph-augmented trees

In the model proposed in this paper, dictionaries will continue to be written in conventional tree-structured XML – or so they will appear to the lexicographers. Behind the scenes, the dictionary writing system will keep track of any relations that 'break out' of the tree and present them to the lexicographer as annotations beside the tree. The rest of this section will show how this approach will alleviate the two lexicographic problems outlined above.

4.1 Placement of multi-word items

An administrator will be able to specify in the dictionary schema which elements in the tree structure can be shared by multiple entries. This will typically apply to phraseological subentries and other multi-word material. When creating a phraseological subentry inside an entry, the lexicographer will be able to create new subentries as normal, but will also be able (and encouraged) to link to existing ones when applicable.

For example, a lexicographer will create the subentry *third time lucky* while working on the entry for *third*. To the lexicographer, it will seem as if the subentry is part of the entry, just like any other XML element. Internally, however, the system will store this subentry separately and link it to the entry for *third*. Later, while working on the entries for *time* and *lucky*, if the lexicographer decides to include *third time lucky* as subentry, he or she will be prompted by the system to bring in the existing subentry instead of creating a new one. Because the subentry is now shared by several entries, any changes made to it will affect all the entries that share it. When editing an entry that contains a shared subentry, the lexicographer will be notified (see Fig. 1) to



```
<entry>
  <headword>time</headword>
  <partOfSpeech>n</partOfSpeech>
  <sense>
    <translation>čas</translation>
    <translation>doba</translation>
    <phraseme>  This subentry also appears under third and lucky.
      <source>third time lucky</source>
      <target>do třetice všeho dobrého</target>
    </phraseme>
  </sense>
</entry>
```

Fig. 1: Notifying the lexicographer of relations that 'break out' of the tree: "This subentry also appears under 'third' and 'lucky'."

make sure they understand that any changes they make to the subentry here will be visible in the other entries too.

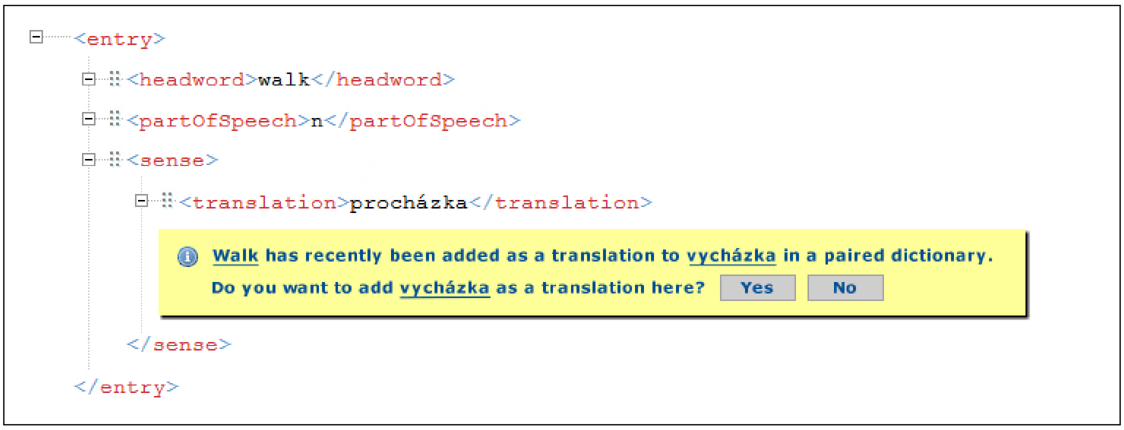
The model proposed here is similar to an approach one often sees in dictionaries where multi-word phrasemes are treated as independent entries, in effect promoting them to the same level as single-word entries. We may call this the 'multi-word promotion' approach. Multi-word promotion solves the problem of phraseme placement by deciding not to place the phraseme anywhere, and that is also its drawback: it strips the lexicographer of the ability to include a phraseme like *third time lucky* in a specific sense of a single-word entry, for example a specific sense of *time*.

The 'sharing' model proposed here is in fact an implementation of a less-known feature of Lexical Markup Framework (LMF) [6] where multi-word entries can be independent entries which can then be linked to from specific senses of other entries via their ID.

4.2 Bilingual dictionary reversal

An administrator will be able to set up a 'mapping' between the schemas of two dictionaries, such as a pair of dictionaries where one goes from language X to language Y and the other from language Y to language X. These dictionaries will then be 'paired'. As lexicographers make edits to entries in one of the dictionaries, the system will keep track of the edits and later suggest corresponding edits to the other dictionary in the pair. For example, when a lexicographer adds the translation *walk* under the headword *vycházka* in one dictionary, the system will remember to suggest adding the reverse translation *vycházka* to the appropriate headword *walk* in the other dictionary (see Fig. 2).

This way the lexicographers will be encouraged to keep the two dictionaries synchronized.



```
<entry>
    <headword>walk</headword>
    <partOfSpeech>n</partOfSpeech>
    <sense>
        <translation>procházka</translation>
    </sense>
</entry>
```

Walk has recently been added as a translation to vycházka in a paired dictionary.
Do you want to add vycházka as a translation here?

Fig. 2: Keeping paired dictionaries synchronized semi-automatically: "'Walk' has recently been added as a translation to 'vycházka' in a paired dictionary. Do you want to add 'vycházka' as a translation here? Yes – No."

The model proposed here is similar to, but subtly different from, the approach sometimes taken by dictionary projects where lexemes exist not as strings but as links to another database. For example, in the Cornetto project [9] there are two databases: a monolingual dictionary and a wordnet. The wordnet does not contain any literal lexemes: instead, it has links to specific senses of specific headwords in the monolingual dictionary. If headwords in the monolingual dictionary are changed or deleted, the changes will be refected in the wordnet automatically.

The 'pairing' model proposed here does not envisage such automation: it does not envisage that changes in one dictionary would be reflected in another dictionary automatically. Instead, the system would only keep track of changes in one place and **suggest** corresponding changes in other places. It would be up to the lexicographer to accept or reject the suggestions. The fact that the pairing is not fully automatic is what, it is hoped, would make this way of working more compatible with how lexicographers usually work: the final content of each and every entry would be the result of a lexicographer's decision, like it always has been in lexicography – except this time the decisions would be 'computer-aided' (consider the analogy of Computer-Aided Translation, CAT, where a software tool suggests candidate translations and a human translator either accepts or rejects them).

4.3 Other benefits of graph-augmented trees

The hybrid data model proposed here has benefits that stretch beyond the two scenarios described above.

The notion of shareable subentries can be used for other entry components besides phrasemes, such as example sentences. A sentence like *who's the lucky winner?* is a good illustrative example for both *lucky* and *winner*. Instead of creating two copies of the sentence in two entries, it could be stored in a single copy internally and **shared** by the entries. Later, if lexicographers want to edit the sentence (say to correct a spelling mistake) or add a translation to it, they only need to do it once, saving work and avoiding any potential for inconsistencies.

The same could even apply to translation equivalents inside senses. In many dictionaries translations are nothing more than strings of text but, in some, translations are decorated with extensive grammatical and other annotations. When the same translation appears under multiple headwords, as they often do, lexicographers' time is wasted entering the same information again and again. Instead, translations could be 'shareable', thus again saving work and avoiding potential inconsistencies.

The concept of paired dictionaries too can be used for other purposes besides bilingual reversal. The paired dictionaries can be related by means other than reversal, for example by the lexicographic function [13] they fulfil, such as the type of their target audience: one can be a beginner's dictionary and the other a larger dictionary for advanced learners of the same language. In such a situation, an entry in the beginner's dictionary is typically an abridged version of its counterpart in the advanced dictionary. When a lexicographer makes an edit to one of the pair, such as add a new translation or an example sentence, the system will remember to propose a corresponding edit in the other dictionary, thus helping to keep the two synchronized.

The notions of 'sharing' and 'pairing' can even be combined into a single setup. For example, a dictionary and a thesaurus of the same language can share definitions, while the system keeps track of paired senses in both.

5 Conclusion

As an industry, lexicography is facing life-changing challenges at the moment. As revenue from commercial dictionary sales is decreasing, lexicography is moving from the private sector to the public sector where it needs to function on limited budgets. In such circumstances it becomes important to be able to 'do more with less': to deliver more dictionaries more quickly, with less effort. The model of graph-augmented trees, if and when it becomes implemented in an industrial-strength dictionary writing system, will empower lexicographic teams to deliver precisely that, while allowing them to continue working within the familiar paradigm of trees. The techniques of 'sharing' and 'pairing' are time-saving devices which remove the need for repetitive data entry and simultaneously ensure greater consistency between individual entries and entire dictionaries.

Acknowledgments. This publication was written with the support of the Specific University Research provided by the Ministry of Education, Youth and Sports of the Czech Republic.

References

1. Aguado-de-Cea, G., Montiel-Ponsoda, E., Kernerman, I., Ordan, N. (2016). From dictionaries to cross-lingual lexical resources. In: Kernerman Dictionary News, 24, pp. 25-31.
2. Atkins, B. T. S., Rundell, M. (2008). The Oxford guide to practical lexicography. Oxford: Oxford University Press.
3. Bogaards, P. (1990). Où cherche-t-on dans le dictionnaire ? In: International Journal of Lexicography, 3(2), pp. 79-102.
4. Fellbaum, C. (1998). WordNet: an electronic lexical database. Cambridge: MIT Press.
5. Granger, S., Paquot, M. (2012). Electronic lexicography. Oxford: Oxford University Press.
6. ISO 24613:2008 Language resource management – Lexical markup framework (LMF). `http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=37327`
7. Ježek, E. (2016). The lexicon: an introduction. New York: Oxford University Press.
8. Klimek, B., Brümmer, M. (2015). Enhancing lexicography with semantic language databases. In: Kernerman Dictionary News, 23, pp. 5-10.
9. Horák, A., Rambousek, A., Vossen, P., Segers, R., Maks, I., van der Vliet, H. (2009) Cornetto Tools and Methodology for Interlinking Lexical Units, Synsets and Ontology. In: Current Issues in Unity and Diversity of Languages. Seoul: The Linguistic Society of Korea, pp. 2695-2713.
10. Maks, I. (2007). OMBI: The Practice of Reversing Dictionaries. In: International Journal of Lexicography, 20(3), pp. 259-274.
11. Měchura, M. (2016). Things to think about when building a dictionary website. Talk at meeting of European Network of e-Lexicography. `http://www.lexiconista.com/things-to-think.pdf`
12. Polguère, A. (2004). From Writing Dictionaries to Weaving Lexical Networks. In: International Journal of Lexicography, 24(7), pp. 396-418.
13. Tarp, S. (2008). Lexicography in the Borderland Between Knowledge and Non-Knowledge: General Lexicographical Theory with Particular Focus on Learner's Lexicography. Tübingen: De Gruyter.