

Tools for upgrading printed dictionaries by means of corpus-based lexical acquisition

Ulrich Heid*, Bettina Säuberlich*, Esther Debus-Gregor°, Werner Scholze-Stubenrecht§

*Institut für maschinelle Sprachverarbeitung, University of Stuttgart, Azenbergstr. 12, D 70174 Stuttgart, Germany
{heid, tina}@ims.uni-stuttgart.de

°Langenscheidt KG, Mies van der Rohe Straße 1, D 80807 München, Germany
e.debus@langenscheidt.de

§Duden BIFAB AG, Postfach 100 311, D 68003 Mannheim, Germany
Werner.Scholze-Stubenrecht@bifab.de

Abstract

We present the architecture and tools developed in the project TFB-32 for updating existing dictionaries by comparing their content with corpus data. We focus on an interactive graphical user interface for manual selection of the results of this comparison. The tools have been developed and used within a cooperation with lexicographers from two German publishing houses.

1. Introduction

1.1. Dictionary Writing vs. Dictionary Updating

Most contemporary dictionaries are corpus-based. Since the Hector project (Atkins 1992), most dictionary publishers have used corpus data for producing monolingual or bilingual print dictionaries. Computational support tools for corpus-based dictionary writing range from KWIC indices, over specific in-house corpus access systems, to corpus digest systems, such as the well-known WASPS tools (Kilgarriff/Tugwell 2001). These tools support the writing of dictionaries from scratch, providing corpus sentences from where information about words or word combinations can be derived.

In practical lexicography, however, much more frequently existing dictionaries have to be updated than new ones written from scratch. In this paper, we describe tools for dictionary updating¹. This task involves not only lexical acquisition from text corpora, but also an analysis of the electronic version of the existing dictionary, and a comparison of linguistic descriptions abstracted from both sources. The tools check which corpus-derived facts about a lexical item are already contained in an existing dictionary and vice versa. For each relevant linguistic property, candidates for inclusion or possibly for removal are suggested, and the lexicographer's task is to select those items that best fit the information programme of the dictionary. Selected items can then be exported to the publisher's dictionary writing system.

In the rest of this introduction, we give an overview of the system (section 1.2.). We then discuss its architecture

(section 2.), as well as dictionary (2.2.) and corpus analysis (2.3.). In section 3., we describe LexiView, a graphical user interface for the lexicographer. Section 4. is devoted to further work.

1.2. System Overview

Figure 1 is a schematic overview of the system. Dictionary and corpus data are its input. Modules for dictionary analysis, as well as for lexical acquisition are used to abstract descriptions of linguistic phenomena from both sources. These are represented in an XML-based internal format to allow a comparison between corpus and dictionary data. The comparison results again represented in XML, is submitted to the lexicographer via the LexiView interface.

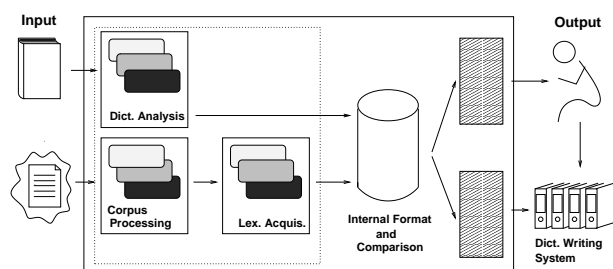


Figure 1: Schematic overview of the TFB 32 system

2. System Architecture

2.1. Representing Language Data from Dictionaries and Corpora

To be able to compare data from a printed dictionary with data extracted from corpus text, a common representation format is needed. This format is general enough to account for all relevant facts about a given word. It contains, however, only those which can be abstracted automatically from corpus data:

¹This work is the result of the project *Transferbereich 32* (TFB 32), a cooperation between Duden BIFAB AG, Mannheim, Langenscheidt KG, München, and the Computational Linguistics department of IMS, University of Stuttgart. The university part of the project was funded by the Deutsche Forschungsgemeinschaft, DFG, in the time frame between 10/2001 and 12/2003.

- lemma and word class (used as an identifier);
- corpus frequencies of a lemma + word class pair;
- linguistic properties of a lemma + word class pair (e.g. syntactic subcategorization, morphosyntax, etc.) and their frequencies;
- collocations and other significant word pairs, and their linguistic properties and frequencies.

Furthermore, (pointers to) example sentences are included. The basic DTD of the internal format is displayed in Figure 2, in a version simplified for readability. The format does however not account for readings, because it is not possible in the general case (only in exceptional cases) to automatically match corpus sentences against semantic readings from a dictionary.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>

<!ELEMENT lexicon (entry+)>
<!ELEMENT entry (HWD, Freq, POS, Inflectn?,
  Examples*, Marking?, Colloc*)>

<!ELEMENT HWD (#PCDATA)>
<!ELEMENT Freq (EMPTY)>
<!ATTLIST Freq absolute CDATA #REQUIRED>
<!ATTLIST Freq ppm CDATA #REQUIRED>
<!ELEMENT POS (#PCDATA)>
<!ELEMENT Examples (#PCDATA)>
<!ELEMENT Inflectn (#PCDATA)>
<!ELEMENT Marking (#PCDATA)>

<!ELEMENT Colloc (Colloc-Example*)>
<!ATTLIST Colloc freq CDATA #REQUIRED>
<!ATTLIST Colloc type CDATA #REQUIRED>
<!ATTLIST Colloc where (dict|corpus|both) #RE-
QUIRED>
<!ELEMENT Colloc-Example (#PCDATA)>
```

Figure 2: Internal format: Parts of the DTD

The internal format is not comparable with standards-oriented formats, such as chapter 12 of the TEI Guidelines: Its objective is not to reproduce dictionary article structure, nor to map a complete dictionary. We aim at extracting data about relevant linguistic phenomena, irrespective of the indication type.

2.2. Dictionary Analysis

Dictionary analysis in the TFB 32 tools is not reformatting, but selective extraction. We keep track of both explicit and implicit indications. Implicit ones, i.e. all example sentences, are treated the same way as corpus sentences. In addition, there is a need to resolve lexicographic text condensation, e.g. the listing of adjectives in *ein interessantes, beliebtes, heikles, aktuelles, politisches, literarisches Thema*. (Duden Universalwörterbuch, 4/2001, s.v. Thema).

The dictionary data were made available as SGML or XML texts by the publishing houses. As each publishing

house has different formats, often even for different dictionaries, each dictionary must be analyzed individually and then mapped to TFB's internal format.

2.3. Corpus Analysis

All lexical acquisition tools can be used for corpus analysis, provided they produce an output compatible with the internal format of TFB 32. Currently, acquisition tools for subcategorization frames, collocations, morphosyntactic properties of words and of collocations (e.g. preferences for singular vs. plural) are in use. They are modules and can be replaced if necessary. They make use of recursively chunked corpora (cf. Kermes 2003) and of stochastic corpus parsing with the Gramotron grammar (cf. Schulte im Walde et al. 2002). These tools are obviously language-specific for German, whereas the internal format abstracts away from individual languages.

Corpus material has been taken from freely available or specifically licensed newspaper texts, among others *Frankfurter Rundschau* (1992/93, from the ECI), *Stuttgarter Zeitung* (1992/93, special license), a total of over 350 million words. Our corpus is not balanced, as a general balanced corpus of German is only being created, e.g. at BBAW². Evidently, certain results of the comparison between corpus and dictionary are relativized by the nature of the corpus used. Even though this has no impact on the tool design, of course all results must be screened by a lexicographer. In this sense, the system is interactive: the lexicographers decide about the comparison results.

2.4. Comparing Dictionary and Corpus

The comparison between corpus and dictionary data is carried out automatically. Comparison criteria are the presence or absence of a given fact in one of the sources, as well as frequency and/or significance. Significance is calculated, for binary word combinations (e.g. collocation candidates) by means of lexical association measures (e.g. the log likelihood ratio test (Dunning 1993), or t-score). Thresholds for the definition of inclusion candidates (most frequent items not covered by the dictionary, down to a certain threshold) and removal candidates (items from the dictionary with a corpus frequency below a certain threshold) can be defined interactively. These figures depend, a.o., on the intended size of the updated dictionary, on the amount of material to be removed, and on other parameters (see section 3.1. for details).

3. LexiView – an interactive GUI

The results of the comparison are loaded to LexiView, for interactive inspection and selection. Before we discuss this tool, we indicate the criteria applied in the interactive selection process.

3.1. Criteria for Inclusion and Removal of Items in Dictionary Updating

The decision about including new words, collocations, linguistic facts about words, or example sentences into a

²http://www.dwds.de/pages/pages_textba/dwds_textba.htm

dictionary must be made by the lexicographer. It cannot be automated, since it depends on many complex criteria. The same holds for the removal of items.

Still, space is important in the production of printed dictionaries: a dictionary should evenly cover the targeted vocabulary. If there were enough space, any linguistic fact could be included. In practice, corpus frequency is only one of several selection criteria. Additional inclusion candidates may come from the publishers' own citation files; another source are consistency checks within the macrostructure (if, e.g., chess pieces are part of the nomenclature, all of them should be included, not just a few). If there are similar entries already in the dictionary, the inclusion would not lead to a gain in information, and the candidate may not be included, even if other criteria would suggest it. The most important criterion is the user perspective: the intended use and user group of a dictionary, the prior knowledge and the needs of the users.

These criteria are not easily formalizable; thus manual selection is vital, and must be supported as much as possible by the user interface, e.g. through access to sublanguage marks or to example sentences from corpus and dictionary.

3.2. Workflows and GUI Principles

The LexiView tool is intended for work at macrostructural and at microstructural level; there are two possible workflows:

- lexicographers may first decide on macrostructural updates (which entries to remove or to include), and then, in a second step, on microstructural updates, i.e. on additional facts about these words;
- alternatively, the lexicographers may make reference to descriptive details at the time of deciding about inclusion and removal of entries, thus dealing with macrostructure and microstructure together.

Both workflows are supported by the three-part layout of the standard LexiView screen. The three parts typically contain the following kinds of data (cf. 3):

- a lemma list with features of each lemma ('Table'-window: with corpus frequency, part of speech, regional or sublanguage use);
- lists of details about the lemma, by types of linguistic phenomena ('Collocations'-window: e.g. collocations, by grammatical type);
- illustrative material from the corpus and/or from the existing dictionary, displayed in individual windows, for each source and/or type ('Examples'-window).

The examples can only be viewed (or copied to an external file), whereas the lemma lists and the lists of detailed linguistic information can be used in two ways: they can be viewed, but it is also possible to select or unselect items by means of checkboxes ("take" and "cons(ider)" in Figure 3). Selected items may be exported to the upcoming dictionary³.

³As the number and interpretation of checkboxes can however be defined by the user, different checkboxes could have different functions within the dictionary-making workflow.

The result of the comparison between corpus and dictionary is signaled in two ways:

- in lemma lists, arrows mark inclusion (blue arrow: →) and removal (red arrow: ←) candidates;
- in collocation lists colours mark the presence in both sources (black), in the dictionary alone (blue) or in the corpus alone (red)⁴.

3.3. Support functions

The default order of data in LexiView is alphabetic, by lemmas. However, all columns may be sorted in ascending or descending order; thus a lexicographer may view a set of items also by frequency, by part of speech, etc. Similarly, all items suggested for inclusion or for removal can be shown together. The original alphabetic order can always be re-established.

Items can be searched (string search). To allow for comments on individual items, one or more free text comment fields are attached to each lemma. Furthermore, certain fields may be edited by the lexicographer, e.g. to correct typographic errors or misclassifications. The users can also define which fields are editable, which layout the tables have (number and type of columns), and how many 'examples'-sub-windows there are.

3.4. Implementation Principles – Flexibility

LexiView is implemented in Java and platform-independent. It operates on XML-encoded data files and is parameterized via a configuration file and a preference file. The configuration file contains user-modifiable links between the XML elements of the lexical data file and the components of each window used in the display. The user may further (re-)name the windows and the column headers and define which type of data the columns may contain (string, integer, checkbox, etc.).

In the preference file, the appearance of the LexiView GUI is stored, e.g. in terms of the order and the width of columns or the colour of the windows. The last resizing is restored after reloading the tool. This allows each lexicographer to determine interactively the layout optimal for a given task.

Due to this flexibility, any kind of linguistic data may be loaded into LexiView, and the number of information types and operations on them can be determined by the user. This is relevant not only for a broad applicability of the tool (it has, e.g., also been used, for manual checking of automatically generated subcategorization patterns, within work towards an NLP dictionary), but also to support flexible workflow design in practical lexicography. For example, by means of a simple preprocessing step applied to candidate data, results from previous selection exercises could be integrated into the workflow: if certain corpus-generated inclusion candidates have been refused earlier, they can be removed or flagged.

⁴As the comparison between corpus and dictionary is carried out before the data are loaded to LexiView, these settings can not be changed interactively.

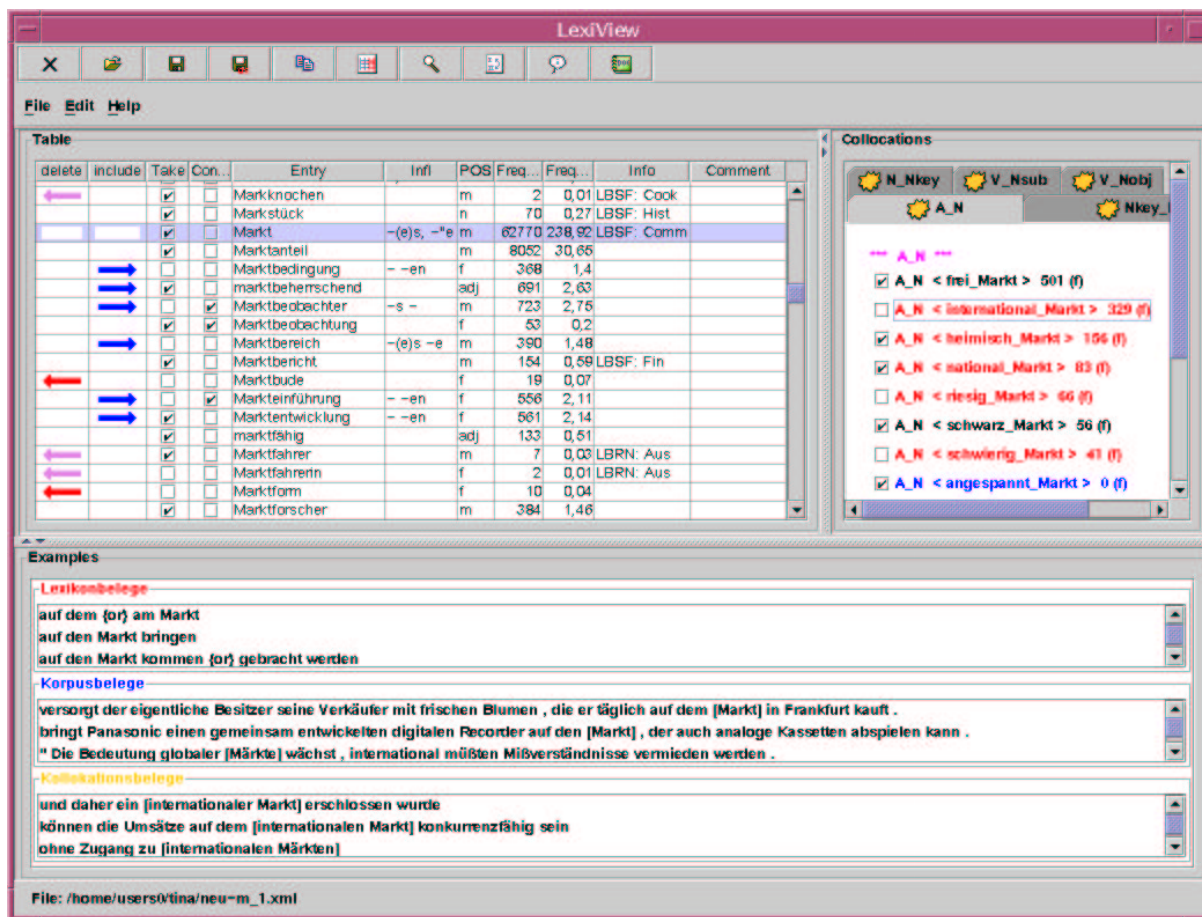


Figure 3: LexiView screen for the word *Markt*

3.5. Integration into Dictionary Production

The results of manual inspection work with LexiView can be exported in tabular or list-like reports or in proto-entries in the format of the publisher's dictionary writing system. For the latter, any applicable XSLT stylesheet can be plugged into the tool at runtime. It is possible to export subsets of the data according to the selection introduced manually: all selected or all unselected items of a selection column (or all items together) may be exported.

4. Conclusions, further Work

We have described the computational linguistic tool setup of the *Transferbereich 32*, aimed at support for the semi-automatic updating of existing dictionaries. Automatic lexical acquisition and a comparison between data from an existing dictionary and from a corpus are combined to provide to lexicographers candidates for the inclusion in or the removal from the dictionary. The manual selection among these candidates is supported by LexiView, a flexible graphical user interface.

Currently, all data are handled in XML files. We plan to support the system by use of a relational database, to store corpus and dictionary data. The comparison between dictionary and corpus will then be carried out by means of user-defined views and filters on database tables.

So far, the tools have been used by Langenscheidt and Duden publishers, as well as within a publishing house out-

side Germany. The current functionality of LexiView is due to a very close collaboration with Langenscheidt lexicographers. The German part of the *Langenscheidt Muret-Sanders Großwörterbuch Deutsch-Englisch* was updated recently with a precursor of the TFB-32 tool suite.

5. References

- Beryl T. S. Atkins: "Tools for computer-aided corpus lexicography: the Hector project", in: *Acta Linguistica Hungarica* 41, 1-4, 1992-93, 5 – 71
- Ted Dunning: "Accurate Methods for the Statistics of Surprise and Coincidence," *Computational Linguistics*, 19/1, 61 – 74.
- Hannah Kermes: *Offline (and Online) Text Analysis for Computational Lexicography*, Diss., Stuttgart, (Stuttgart: IMS), AIMS
- Adam Kilgarriff, David Tugwell: "Word Sketch: Extraction and Display of Significant Collocations for Lexicography". In: *Proceedings of the workshop "Collocation: Computational Extraction, Analysis and Exploitation"*, 39th ACL & 10th EACL, Toulouse, July 2001, 32 – 38.
- Sabine Schulte im Walde, Helmut Schmid, Mats Rooth, Stefan Riezler, Detlef Prescher: "Statistical Grammar Models and Lexicon Acquisition", in: Christian Rohrer, Antje Roßdeutscher, Hans Kamp (Hg.): *Linguistic Form and its Computation*, (Palo Alto, CA: CSLI Publications), 2001: 387 - 440.