jaSlo, a Japanese-Slovene Learners' Dictionary: Methods for Dictionary Enhancement

Tomaž Erjavec*, Kristina Hmeljak Sangawa, Irena Srdanović Erjavec****
*Department of Knowledge Technologies, Jožef Stefan Institute
**Faculty of Arts, University of Ljubljana
tomaz.erjavec@ijs.si, kristina.hmeljak@guest.arnes.si, irena_srdanovic@hotmail.com

Abstract
The paper presents our experiences in producing the hypertext learners' Japanese-Slovene dictionary jaSlo, which currently contains over 10,000 entries. The paper discusses the conversion of the dictionary from the legacy encoding, which consisted of many separate files in a mixture of different tabular formats, into a standardised XML format. The conversion consisted of uptranslation from the legacy formats, the enrichment of the dictionary with third-party resources, merging of the data, manual verification, and the deployment of the dictionary via a Web interface, available at http://nl.ijs.si/jaslo/. The presented methodology ensures that the resulting dictionary is of a high quality, addresses user's needs, and is suitable for re-purposing and interchange. We conclude with plans for further work.

1 Introduction

The establishment of a new course of Japanese studies at the University of Ljubljana in 1995 brought with it the need for Japanese dictionaries for Slovene speaking students. However, due to the limited number of potential users, probably not much more than the current 180 students of Japanese at the department, the compilation of such dictionaries is not a particularly profitable project that could interest a publishing house. The teachers at the department therefore decided to create it with the help of our students, the final users of the dictionary (Hmeljak Sangawa, 2002).

The compilation of a dictionary that would satisfy all the needs of our students is going to last for many years. Meanwhile, even incomplete data can be useful to users of a language pair for which no dictionary exists at all. We therefore decided to merge the various glossaries created at our department and publish them on the web.

The first stage of converting an initial dictionary (1000 entries in tabular format) into XML was reported in Erjavec et al. (2004). The target encoding takes into account international standards in the field, which brings with it a number of well-known advantages, such as better documentation, the ability to validate the structure of the document, simpler processing, easier integration into software platforms, longevity and easier Web deployment. In this paper, we discuss the second stage of the project, where the dictionary, named jaSlo, was expanded to contain over 10,000 entries, normalised and enriched by various third party resources. The focus of the paper is on presenting the methodology used in producing this new dictionary, which could also benefit other similar collaborative lexicographic projects.

2 Dictionary encoding

For encoding the dictionary we used the XML version of the Text Encoding Initiative Guidelines, TEI P4 (Sperberg-McQueen & Burnard, 2002), in particular its module for dictionary encoding.

```
<entry id="jaslo.6557">
<form type="hw">
<orth type="kana">ちょうせつする</orth> <orth type="kanji">調節する</orth>
<orth type="roma">chousetsusuru</orth>
</form>
<gramGrp><pos>Vs</pos> <subc>trans.</subc></gramGrp>
<trans><tr>uravna(va)ti</tr></trans>
<eg>
<q>室内（しつない）の温度（おんど）をちょうせつする</q>
<tr>uravnavati temperaturo v sobi</tr>
</eg>
<xr type="lesson" n="L1.23"><xref>1. letnik, lekcija 23</xref></xr>
<usg type="level">0</usg>
<note type="admin" resp="TER">2005-07-11 Add romaji</note>
<note type="admin" resp="TER">2005-07-10 Add levels</note>
<note type="admin" resp="ISE">2005-02-28 Merge</note>
<note type="admin" resp="VOJ">2005-02-22 V (440)</note>
<note type="admin" resp="KHS">2003-03-12 L1 (850)</note>
</entry>
```

Figure 1. A typical dictionary entry in jaSlo

Figure 1 presents a typical dictionary entry, which includes the form of the headword given in kanji, kana (hiragana or katakana) and in Latin transcription, so called romaji. This is followed by grammatical information, translation into Slovene, examples, a reference to the lesson where the word is introduced, the difficulty level of the entry, and finally administrative information tracing the compilation history. In addition to the elements given in the example, the following information is also present in a subset of the entries: cross-reference to related entries (esp. synonyms with different levels of politeness etc.), inflected forms of verbs, the etymology of loan-words, and encyclopaedic descriptions of proper names and Japanese culturally bound terms.

3 The Compilation Process

3.1 Up-translation

The process of up-translation to the standard TEI encoding had to cope with a plethora of file and input formats, some of them containing implicit structures, and was implemented using Perl.

For up-translation of the tabular dictionaries, the source character encoding was first converted from Shift-JiS to UTF-8, and the files then converted to TEI. The transformations, for most fields, simply wrapped their content into the appropriate TEI tags. Additionally, the programs also performed some normalisation (e.g. stripping superfluous whitespace and punctuation, normalising variant spellings of labels), verification (e.g. detecting illegal empty fields and flagging suspicious elements with a question mark) and assignment of tags ac-

cording to detected string patterns, explicitly marking information that was implicit in the original format. So, for example, the note column of some source files contained remarks on usage, but also the etymology of borrowings. Where the pattern »(iz ... ...)« was found, e.g. "*(iz nemšč. Arbeit)*" ("*from German Arbeit*") this was converted to <etym><lang> nemšč. </lang> <gloss>Arbeit</gloss></etym>.

As was seen in Figure 1, each entry also contains administrative notes on the history of the entry – which legacy file it was derived from, when it was created or modified, who modified it, and what the modification was. Such a revision history significantly helps in debugging the transformations as well as giving per-entry author attribution.

Writing the up-translation programs was a long, and, to an extent, frustrating task, familiar to anyone who has attempted to automatically clean and flag 'dirty' data. It consists of a cycle where a transform is written, run over the input, the results evaluated, the transform modified, and the process repeated, all the time striving to find a balance between the precision and recall of the filter. The process typically terminated when we judged that the effort to further modify the program would exceed the effort to manually verify and correct the actual XML dictionary.

3.2 Adding external information

The dictionary entries were also automatically enriched or normalised via (mostly) third party resources, in particular, the following: the transcription of the hiragana headword into the Latin alphabet (romaji); the difficulty level of the headword; part-of-speech normalisation; and the addition of the caron diacritic to Slovene characters. We present these in turn.

Japanese has a very complex writing system consisting of Chinese kanji characters and the phonetic (syllabic) scripts katakana and hiragana. For our dictionary it was initially decided that the primary headword of each entry should be in hiragana or katakana, as in traditional Japanese dictionaries. Entries are accompanied by their kanji (or mixed kanji-kana) orthography when appropriate. However, log files of the usage of the first version of the dictionary showed that users often input Japanese search strings using the Latin alphabet. We therefore added Latin transcription to all entries using a freely available kana to romaji converter program (http://raa.ruby-lang.org/project/kana2rom/). Next, we marked all headwords according to the 4 difficulty levels of the vocabulary list used by the Japanese Language Proficiency Test.

As the legacy tabular files were very inconsistent with regard to part-of-speech, we spent quite some time first devising the PoS set to be used, and then semi-automatically converting the legacy PoS labels to this common standard. Our set of categories contains 19 different labels and is based on the set used in the Japanese morphological analyzer Chasen (Matsumoto et al., 2003). We first normalized obvious mismatches with Perl, then manually assigned canonical PoS to the list of all remaining PoS labels appearing in the dictionary, and used this mapping to correct the source dictionary. As a side benefit, we included the mappings into the jaSlo TEI header; this enables flexibility in the display of the dictionary.

Finally, there was the problem of č, š, ž (and their upper case equivalents), which are the only three characters used for Slovene which are not in the ASCII character set. These char-

acters are also not part of Shift-JIS, the encoding used in most of the legacy dictionaries, where they were typically substituted by c, s, z. Reversing this simplification is unfortunately impossible with automatic means. Although more sophisticated algorithms exist for automatic diacritic insertion (e.g. Tufiş and Chiţu, 1999) we chose a relatively simple method, where each Slovene word containing one of these characters was matched against a large dictionary; if it was found to correspond to an unambiguous dictionary word it was replaced; if it was not found, or was ambiguous, e.g. *resen (serious)* vs. *rešen (saved)* it was flagged for manual verification.

3.3 Merging the data

Each piece of the newly acquired data had to be merged with the evolving dictionary. This procedure consisted of first identifying whether a new entry was being added or an existing one (and which) modified, and, in this case, how to add the new information to an existing entry. The identification of the entry is complicated by the fact that its unique key would have to be a combination of the kana headword string, the kanji, and the part-of-speech (e.g. かく/書く/V "write" vs. かく/欠く/V "be missing"); if any of these fields is missing from the key, we can be faced with ambiguities. As we, at the outset, did not have a consistent set of PoS categories, and entries could have missing kanji information, the merge program identified ambiguous situations and flagged these for manual verification. The problem was not marginal as the legacy dictionaries had a significant amount of overlap in contained entries.

However, it was, in general, not possible to simply discard duplicate entries, as they could each contain valuable information, e.g. examples, reference to the lesson number where the word is introduced etc. The merge therefore identified several possible situations. When the information in the new entry was already contained in the dictionary, the new information was simply ignored; when the information from the two sources could be unified monotonically, and the two entries matched in the complete key, the new information was straightforwardly added to the existing entry. When, however, the entries had incompatible information or they did not match in the PoS, all entries were wrapped in a <hom> (=homonym) element, with its n attribute giving the number of 'homonymous' entries, and these were then merged manually.

3.4 Manual verification

The general method of producing the dictionary involved programs that had a less than perfect precision but did strive to identify dubious cases and flag them for manual verification. This involved adding question marks to suspicious element content or, in the case of merging, the use of an extra tag. For easier handling and to enable simultaneous work, the automatically produced dictionary was split into 11 files and these were then manually verified with the help of an XML editor.

4. Using the dictionary

The dictionary is deployed via a Web-based interface, available at http://nl.ijs.si/jaslo/, which allows full text searches by string or word on the dictionary, with optional restriction

of the match to headword or translation, and filtering by PoS or difficulty level. The interface is also localised to Slovene, Japanese and English. The user's browser is assumed to offer Unicode support and have installed a Japanese-language font but, apart from that, no requirements are imposed on the client architecture. The server is implemented as a Perl CGI script, which accepts the search parameters and sequentially, via a SAX filter, returns the entries that match the query, using an XSLT stylesheet similar to the one used by the editors, but which ignores certain information, e.g. admin notes, entry ID etc. While this means that for each query the complete dictionary has to be processed, this does not present problems with the current size of the dictionary and user load.

Each query together with time and number of returned entries is also logged (without client machine address, thus preserving privacy), which enabled us to begin tailoring the dictionary to user needs. Aside from romaji transcriptions, we noticed heavy use of searching by lesson number only, i.e. the students obviously find it convenient to extract from the dictionary the complete set of entries introduced in a given lesson. The log file will also come in useful for further expansion of the dictionary, by isolating the most frequently searched-for but not found words.

5 Conclusions

"Collaborative bottom-up editing" and open-source lexicographical projects have been criticized (see e.g. Docherty, 2000) for their poor quality, which is indeed often the case. However, collaborative editing can produce useful data and may be the only viable means of producing a dictionary for a non-profitable language pair. Looking back we can conclude that it would have been well worth investing time up-front to specify precise guidelines for dictionary encoding, to use a platform that prevents syntactically ill-formed input, and to coordinate the dictionary making activity to prevent duplication. Still, for others in a similar situation, i.e. faced with varied and inconsistent legacy data, we believe that our approach presents a viable method. The approach is predicated on the use of open platforms and tools (Linux, Apache, Perl), standards (XML, TEI, XSLT, HTML), and on the use of supplementary resources (Slovene lexicon; kana2rom, Chasen) and consists of an uptranslation, followed by a merge operation, manual post-editing, and Web deployment, at http://nl.ijs.si/jaslo/.

There are a number of improvements to jaSlo we are planning in our future work. For further additions as well as corrections to the dictionary we will implement a Web-based form interface, with a human editor checking the proposed updates prior to incorporation into the master dictionary. An interesting venue of further work is also to enrich the dictionary searching and display by automatically creating links between the dictionary and a kanji database and to external Web dictionaries, e.g. WWWDict (Breen 2003), one of the best-known Japanese-English Web dictionaries, or the Slovene-German-Slovene dictionary for German students of Slovene (Lönneker and Jakopin, 2003), a project similar to ours. We are also planning to add jaSlo into the on-line Japanese reading support tool "Reading tutor" (Kawamura et al., 2003).

References

A. Dictionaries

Breen, J. (2003), The Japanese-Multilingual Dictionary and the Japanese Proper Names Dictionary. http://www.csse.monash.edu.au/~jwb/japanese.html

Kawamura, Y., Kitamura, T., Hobara, R. (1997-2002), Reading Tutor. http://language.tiu.ac.jp/

B. Other Literature

Docherty, V.J. (2000), 'Dictionaries on the Internet: an Overview', in *Proceedings of the Ninth Euralex International Congress, Euralex 2000*, pp.67-74, Stuttgart.

Erjavec, T., Hmeljak Sangawa, K., Srdanović, I., Vahčič, A. ml. (2004), 'Making an XML-based Japanese-Slovene Learners' Dictionary', in *The 4th International Conference on Language Resources and Evaluation (LREC) proceedings, ELRA*, pp. 1059-1062, Lisbon, 2004.

Hmeljak Sangawa, K. (2002), 'Slovar japonskega jezika za slovenske študente japonščine' (A Japanese Dictionary for Slovene Students of Japanese), in *Proceedings of the Conference on Language Technologies*, pp. 102-105, Ljubljana: Jožef Stefan Institute.

Lönneker, B. (2003), 'The Slovenian-German-Slovenian online dictionary for learners: Current status', in *International Journal of Lexicography 16/4. (EURALEX newsletter)* 462-463.

http://webapp.rrz.uni-hamburg.de/~slowenisch/

Matsumoto, Y. et al. (2003), Morphological Analyzer Chasen. http://chasen.aist-nara.ac.jp/

Sperberg-McQueen, C. M., Burnard, L. (eds.) (2002), Guidelines for Electronic Text Encoding and Interchange, The XML Version. The TEI Consortium. http://www.tei-c.org/

Tufiş, D., Chiţu, D. (1999), 'Automatic Diacritics Insertion', in *Romanian Texts. Proceedings of COMPLEX'99 International Conference on Computational Lexicography*, Pecs.