Linking Linguistic Resources: time aligned corpus and dictionary

Michel Jacobson, Boyd Michailovsky
LACITO, CNRS, Villejuif, France
jacobson@idf.ext.jussieu.fr, boydm@vjf.cnrs.fr

We present preliminary results in linking computerized, multimedia speech documents of the LACITO Archive project to a computerized dictionary. The speech documents are time-aligned with recordings and have a structure defined by an XML DTD, which has been presented elsewhere (Michailovsky 2001, Jacobson and Michailovsky 2000). Over 70 of these documents, including the sound, may be consulted on the LACITO Archive Project site (http://www.lacito.archivage.vjf.cnrs.fr.) The dictionary we will start with is a Limbu-English bilingual dictionary originally developed in a plain-ASCII structured format for use with Robert Hsu's LEXWARE suite of modules for lexicography, and recently converted to a TEI-inspired XML format. Limbu is a Tibeto-Burman language of Eastern Nepal.

A basic design philosophy of the LACITO Archive project has been to keep the markup of speech documents simple, or at least to allow for simple markup. This is to make it easy for researchers to mark up large amounts of text, perhaps reserving more detailed markup for a few demonstration texts or texts of particular interest. To compensate, we would like to be able to link items in running text to dictionary entries, which in our view is where a lot of the detail belongs, although these, too, may start out simply. The dictionary entries supply lexical information that is not in the text markup -- and which should not have to be repeated every time a word occurs in a corpus. Further, we would like to see how far we can get with automatic linking -- that is, without having to hand-lemmatize text items -- even if it means that not all items link correctly and unambiguously to dictionary entries. A background assumption is that many linguists will simultaneously be working on texts and dictionaries.

1. Dictionary format

In designing the dictionary format, we have adopted a number of structures and labels from the TEI guidelines for dictionaries (Sperberg-McQueen and Burnard 2001: ch. 12) as a starting point, without trying to use the TEI DTD, which is very complex. These guidelines -- as can be inferred from the chapter heading "Print Dictionaries" -- are not designed for linguists' dictionaries of spoken languages; they are intended for marking up traditional, printed dictionaries. They define an XML markup which can be built upon and transformed if necessary using standard tools. We have added markup to cover additional data categories important to field linguists and functions relevant to online as opposed to printed material. The current dictionary markup is an early draft (comments welcome).

LEXWARE

We return briefly to our original starting point, the LEXWARE-formatted dictionary (Hsu 1989). The format is quite similar to the SIL "standard format" used by SHOEBOX and MDF (anon. ?2000, Coward and Grimes 2000). The LEXWARE suite provides modules for managing and exploiting data in this format. (Mechanisms for repeating blocks of fields (e.g. 1def - 1example, 2def - 2example), for subentries, and other refinements are not illustrated here.)

```
.lim    tumba
poss    kundumba
fem     tumma
dial    libang
ps      n
edf     *elder, eldest *brother of a sib; elder *uncle or *aunt ;(father's elder brother, mother's elder sister, or their
spouses)
sem     par
par     FeB
par     MeZH
cfetym  tumma%2
lexx    tumba <Y>
```

The above can be marked up in RTF by a LEXWARE module and printed (omitting some fields):

tumba *poss:* **kundumba**. *fem:* **tumma**. *n,* elder, eldest brother of a sib; elder uncle or aunt (father's elder brother, mother's elder sister, or their spouses). *etym:* **tumma^{2}**.

A second module produces an English index with entries like the following:

brother: bond-friend's elder brother	phuʔiŋ
clan-brother	sɔmma[1]
elder brother	phuʔ
elder, eldest brother of a sib	tumba

TEI-based XML

The dictionary has been converted LEXWARE to a TEI-based XML markup using a script. The markup for main entries <entry> provides for five top-level constituents:

- `<form>`: spoken and morphophonemic forms; orthography if available.
- `<usg>`: usage: dialect, level of language, etc.
- `<gramGrp>`: grammatical information (part of speech, etc.)
- `<sense>`: definitions, keys for inverting the dictionary, example sentences (?or references thereto), encyclopedic information, certain semantic categories...
- `<xr>`: Internal and external references.

Only the `<form>` group is necessarily present in every article; the first `<form>` group in an article necessarily contains a `<phon>` element (i.e. a form in our basic phonological transcription) with `type="headword"`. The following is the same dictionary article as above, marked up in XML:

```
<entry id="tumba">
      <form>
            <phon type="headword">tumba</phon>
            <phon type="poss">ndumba</phon>
            <phon type="fem">tumma</phon>
      </form>
      <usg>
            <dial>Libang</dial>
      </usg>
      <gramGrp>
            <pos>n</pos>
      </gramGrp>
      <sense>
            <def>elder, eldest brother of a sib; elder uncle (father's elder
                  brother, mother's elder sister's husband)</def>
            <invertkey>elder</invertkey>
            <invertkey>brother</invertkey>
            <invertkey>uncle</invertkey>
            <sem>par</sem>
            <def type="par">FeB</def>
            <def type="par">MeZH</def>
      </sense>
      <xr>
            <ptr type="cfetym" target="tumma_2">tumma</ptr>
            <xptr type="lexx" target="Y">tumba</xptr>
      </xr>
</entry>
```

Notes:

`<entry id="---">`: `id` is a unique identifier which can be pointed to by an `idref` (see below).

`<phon type="poss">`: allomorph appearing with a possessive prefix.

`<invertkey>`: the key under which the definition appears in the English index.

`<sem>`: semantic class, a limited inventory for certain domains only.

`<def type="par">`: a kinship code; `type="binom"` : a Linnaean binomial, etc.

`<ptr>`: cross-reference to another entry in the dictionary.

`type="cfetym"` : an etymological reference (here to a verb 'to be mature').

`target` in this element is defined as an `idref` in the DTD;

this means that its value must correspond to the `id` attribute of an `<entry>` or parsing will fail.

`<xptr>`: reference to an external item, in this case a printed document. The relevant material, which is an headword in a printed, non-XML dictionary, cannot be linked to, so it is entered as the content of the element.

`type="lexx"`: indicates that the reference is to a printed dictionary

`target="Y"`: identifies the referenced dictionary.

As a second example, here is a simple entry for a verb:

> **cumma** (√**cups-**) *vi,* to assemble (intr.), to gather. • **anigɛ kak cupsigɛ-aŋ cum-dzum cumluŋ cogigɛ** – we^{pe} all gathered and had a council among friends. *fam:* **cups-/sups-**. *in:* **cum**, **cumluŋ**.

```
<entry id="cumma">
      <form>
            <phon type="headword">cumma</phon>
            <phon type="root">cups-</phon>
            <phon type="prstem">cum</phon>
            <phon type="pastem">cups</phon>
      </form>
      <usg>
            <dial>Libang</dial>
      </usg>
      <gramGrp>
            <pos class="v">vi</pos>
      </gramGrp>
      <sense>
            <def>to assemble (intr.), to gather</def>
            <invertkey>assemble</invertkey>
            <invertkey>gather</invertkey>
            <eg>
                  <q>anigɛ kak cupsigɛ-aŋ cum-dzum cumluŋ cogigɛ</q>
                  <trans>
                        <tr> we [PE] all gathered and had a council
                              among friends</tr>
                  </trans>
                  <xptr target="oai:lacito:LIM:TRADE" path="//s[345]"/>
            </eg>
      </sense>
      <xr>
            <wordFamily type="cfv" family="cups-/sups-"/>
            <ptr type="cfin" target="cum">cum</ptr>
            <ptr type="cfin" target="cumluN">cumluŋ</ptr>
      </xr>
</entry>
```

Notes:
`<phon type="root">`: an internally constructed morphophonemic root form (the headword is the infinitive).
`<phon type="prstem">`: a verb stem allomorph.
`<eg>`: illustrative example, including the citation `<q>`, translation `<tr>` and source `<xptr>`.
`<xptr>` : external reference to the source of the illustrative example. In this case, the reference is to parsable (that is, XML-coded) data, namely a text in the LACITO archive, here identified by its Open Archive Initiative identifier. (In fact the particular text linked to in this example is not yet catalogued, so please don't look for it.) This pointer will be converted to a link from the dictionary entry to the corpus.
`<wordFamily>`: a word-family of roots to which the entry belongs.
`<ptr type="cfin">`: cross-reference to the `id` of another entry derived from or containing the current entry.

2. Text markup

In the present case, the original data entry is in a broad phonetic transcription, with linefeed-delimited utterances, space-delimited words, and hyphen-delimited morphemes (except for verb stems, which are marked off by parentheses from tense/agreement affixes). Consider the following text entered by the linguist:

> **ku-mba, ku-ndumba-haʔ mɛ(dzups)ɛ.**

This is converted to XML and to Unicode by a script, which in addition to the separators knows a few things about Limbu morphophonemics. Free translation and synchronization data are prepared separately and merged in. Glosses are omitted in the example. Segmentation is at the levels of utterance (`<S>`), word (`<W>`), and morpheme (`<M>`).

```
<S id="toto4">
      <FORM>kumba, kundumbahaʾ mɛdzupsɛ</FORM>
      <TRANSL xml:lang="en">His father and his uncles gathered.</TRANSL>
      <AUDIO start="15.9840" end="17.5031"/>
      <W>
            <M><FORM>ku</FORM></M>
            <M><FORM>pa</FORM></M>
      <W>
      <W>
            <M><FORM>ku</FORM></M>
            <M><FORM>tumba</FORM></M>
            <M><FORM>haʾ</FORM></M>
      </W>
      <W>
            <M type="vprefix"><FORM>mɛ</FORM></M>
            <M type="pastem"<FORM>>cups</FORM></M>
            <M type="vsuffix"><FORM>ɛ</FORM></M>
      </W>
</S>
```

3. Matching

In linking from the text to the dictionary, we look for an exact match between morpheme-level elements in the text and elements in the dictionary. In the case of Limbu, this has required some work on both sides of the equation. Morphophonemic alternants of the headwords have been handled by either or both of two methods: (1) by supplying a lemmatized morphophonemic form in the morpheme-level text transcription and (2) by adding alternants directly as `<phon>` elements in the `<form>` group of the dictionary entry, and matching text items with these as well as with the headwords. Thus, for example, morphophonemic **tumba** (corresponding to phonological **ndumba**, which appears in the utterance-level transcription) matches the dictionary headword directly. Morphophonemic **cups** (corresponding to phonological **dzups**), on the other hand, does not match the dictionary headword, which is an infinitive; it matches the past stem form, which has been added to all verb entries in the dictionary precisely for this purpose.

Our design aim has been to make the matching process itself as generic as possible. Ideally, we would like to match items according to the criterion of identity of form, and, where morpheme-class information is provided in the text markup, identity of morpheme-class. The former -- identity of form -- is relatively simple for Limbu, since both text and dictionary use the same transcription. If this had not been the case, we would have had to provide for a corpus-specific string transformation as part of the matching process. As for morpheme-class, as we have mentioned, only very limited information is coded in the text markup. Our solution has been to match items belonging to classes which are identified in the text markup with items of the same classes in the dictionary, and items of other classes, which are not marked up in the text, with items belonging to all other classes in the dictionary. Just how generic we can make the matching system will only become clear when we have worked with a variety of corpora and dictionaries in different languages.

The details of what has to be done to make matching possible, and how much can be done automatically, are of course language-specific. For Limbu verbs, the original idea was to match the morphophonemic root, which in a dictionary for linguists would be the headword. But because it is often impossible to derive the root from the stems which are found in texts, this would have required explicit, hand-lemmatization (e.g. `<M type="pastem" root="cups-">`) of every verb stem in the texts. Instead, since the roots had already been entered in the dictionary, and it is (almost) always possible to generate the stems from the root, a script was used to generate extra elements in the dictionary containing the past and present stems. These elements (and only these) are the targets of matches from `<M type="XXstem">` elements in the text. Automation comes at a price: if the stems had been hand-lemmatized they would link infallibly to the correct verb, whereas automatic linking may bring up more than one candidate.

In the Limbu text markup, the classes identified are (1) personal agreement prefix strings, (2) personal agreement suffix strings, (3) present stems, (4) past stems, and (5) preverbs (formants of compound verbs) -- that is, verb stems and finite morphology. Nouns, adverbs, other affixes, clitics, etc., are identified in the dictionary but not in the text markup. In fact, the prefix and suffix strings and their allomorphs were not originally included in the dictionary — they were added to the linked version.

At present, certain forms in the transcription cannot be matched to dictionary entries. In particular, national-language loanwords would need to be looked up in a Nepali dictionary, which we do not have. These items are identified as Nepali in the text markup, a fact which can be used to prevent their being looked up and producing a spurious match. Later it may permit looking them up in an appropriate source. Multi-word or multi-morphemic lexemes in the dictionary are not accessed directly but by references in the entries for the individual morphemes. Personal and geographic names are not identified as such in the text, and do not appear in the dictionary either, so they lead to matching failures. Conversely, as in the case of verb stems (above), some morphemes match more than one dictionary entry. Naturally, the more lexical information is coded in the text document, the easier it is to select the correct dictionary entry.

4. User interface and implementation

The initial user interface is a display of the text in which each morpheme is an anchor, ready to generate a hypertext link to the dictionary. When the user selects a morpheme, the corresponding dictionary entry or entries are displayed in a separate window.

As in the rest of the Archive project, the key to implementation lies in a servlet, which calls an XSLT processor to apply XSLT stylesheets to XML documents (texts and dictionary) and directs the output. The stylesheets produce the HTML which is displayed on the client machine. The remainder of this section describes the corpus-to-dictionary links and how they are produced by a stylesheet; it assumes some familiarity with XSLT and HTML.

The stylesheet that is applied to the XML text document to produce the initial display includes a rule defining each morpheme in the text as an HTML anchor, ready to generate a link back to the servlet:

```
<xsl:template match="M">
        <a href="/xslpgm?lg={@xml:lang}&amp;type={@type}&amp;form={./FORM/text()}">
        <xsl:apply-templates/>
        </a>
</xsl:template>
```

For example, the anchor corresponding to the morpheme cups (above) looks like this:

```
<a href="/xslpgm?lg=x-sil-LIF&type=prstem&form=cups">cups</a>
```

Once activated, the link passes the parameters corresponding to the morpheme to be searched for back to the servlet. The servlet again calls the XSLT processor, this time with the XML metadata and an XSLT stylesheet which defines dictionary lookup and display. This stylesheet contains an xpath expression which, instantiated with the appropriate parameters, looks like this (approximately):

```
document('dicoLimbu.xml')//entry[(./gramGrp/pos='vi') or (./gramGrp/pos='vt')]
[./form/phon[@type='pastem']='cups']
```

The corresponding articles are found in the dictionary, formatted by the stylesheet, and returned to the client machine for display. Any cross-references (<ptr>) in the dictionary are coded as hypertext links which can be activated by the user. These include cross-references between articles in the dictionary, and, where possible, references from example sentences in the dictionary to their source in the corpus.

References

anon. ?2000. The Linguist's Shoebox: Tutorial and User's Guide. SIL International. Waxhaw, N. C. (http://www.sil.org/computing/shoebox/ShTUG.pdf)

Coward, David F. and Charles E. Grimes. 2000. Making Dictionaries: A guide to lexicography and the Multi-Dictionary Formatter. SIL International. Waxhaw, N. C. (http://www.sil.org/computing/shoebox/MDF_2000.pdf).

Hsu, Robert. 1989. Lexware Manual. Second Edition. Linguistics Dept., University of Hawaii. Honolulu.

Jacobson, M. and B. Michailovsky. 2000. A Linguistic archive on the web. IRCS/ISLE/Talkbank Linguistic Exploration Workshop: Web-Based Language Documentation and Description. Philadelphia. December 2000. (http://www.ldc.upenn.edu/exploration/expl2000/papers/michailovsky/index.htm)

Michailovsky, B. 2001. The LACITO Archive project markup. Linguist List Workshop: The Digitization of Language Data: The Need for Standards. Santa Barbara, June 2001. (http://linguist.emich.edu/~workshop/markup/lacito/Lmarkup.pdf)

Michailovsky, B. to appear [2002]. Limbu-English Dictionary of the Maiwa-Mewa Khola dialect, with English index. Mandala Book Point. Kathmandu.

Sperberg-McQueen, C.M. and Lou Burnard, eds. 2001. Text Encoding Initiative: The XML Version of the TEI Guidelines. TEI Consortium. (http://www.tei-c.org/P4X/)