Processing the Text of Bilingual Print Dictionaries

Sean Crist, Ph.D.
Nuance Communications, Inc.
kurisuto at panix dot com

ABSTRACT

Two dictionaries were tagged using Conditional Random Fields. Beyond a fairly low number of training tokens, additional training data does not greatly improve accuracy, but the beginning of the plateau depends on the complexity of the dictionary. Additionally, 100 dictionaries were systematically surveyed to create an inventory of features, stated in terms of frequency across dictionaries; these findings can be used to design reusable dictionary processing tools. Finally, an argument is made against the commonly held conception that dictionary entries can be categorized as either 'regular' or 'irregular', and the implications for dictionary processing are explored.

Keywords

Conditional Random Fields, dictionaries, markup, lexicons

Foreword

"Natural language dictionaries seem like obvious candidates for information management in data base form, at least until you try to do one. Then it appears as if the better the dictionary in terms of lexicographic theory, the more awkward it is to fit relational constraints. Vest pocket tourist dictionaries are a snap; Webster's Collegiate and parse dictionaries require careful thought; the Mel'chuk style of explanatory-combinatory dictionaries forces us out of the strategies that work on ordinary data bases." (Grimes 1984 [9])

1. TAGGING BILINGUAL DICTIONARIES WITH CRFS

Constructing machine-readable natural language resources by hand is labor-intensive and expensive. Text dictionaries contain a tempting wealth of linguistic data, and they appear (at least upon casual inspection) to verge on regularity in their structure. The two considerations have led investigators to repeatedly return to the problem of extracting machine-readable lexical data from human-readable dictionaries.

This problem has given rise to a rich literature running from the late 1970s to the present. There have been several approaches to the problem, to be reviewed below in section 3. There are also areas of frustration which have been a running theme in this literature. Indeed, some authors have expressed skepticism about the value of extracting NLP resources from human-readable dictionaries (see, for example, the remarks by van der Eijk et. al. p. 53-4 [27].) However, the problem continues to attract interest, and new technologies have been applied to the problem as they have become available.

The current work explores the use of Conditional Random Fields (CRFs) in tagging tokens within entries in bilingual dictionaries. The task is to tag each token with a field category such as "headword", "pronunciation", or "etymology". This problem is similar to part-of-speech tagging in natural language text.

The main focus in this article is on practical considerations. How many tokens of training data must be hand-tagged to achieve reasonably usable results? What sort of accuracy can be expected? To find answers to this question and others, this article includes two case studies where CRFs were used to tag dictionary text. One of the dictionaries is fairly simple in its entry structure, and the other is very complex.

In addition to the discussion on CRFs, this article contains two additional major sections. In Section 2 below, I will discuss the results of a systematic survey of 100 bilingual dictionaries. This survey counts how often certain attributes are found across dictionaries, with an eye toward practical implementations and reusability of tools. Section 3 is a general discussion of various topics in the processing of bilingual dictionary text.

1.0.1 Shortcomings of Hidden Markov Models

Over the past two decades, Hidden Markov Models have been applied with considerable success to many types of problems involving the classification of discrete elements which are arranged into sequences.

HMMs continue to be useful, but for certain kinds of application, HMMs have a shortcoming which can be illustrated by reference to the problem of named entity tagging. Consider the following two sentence fragments (both were taken from Wikipedia). A "1" over a token indicates that the token is a part of a name; a "0" indicates that the token is not part of a name.

Unpublished draft. Comments welcome.

Example 1:

	0	0	0	1
...	visited	by	President	George

1	1	0	0	
W.	Bush	during	his	...

Example 2:

1	1	0	0	0	
Zali	Steggall	announces	her	engagement	...

The name *George W. Bush* occurs frequently in a wide range of texts, and is therefore fairly likely to appear in any reasonably balanced corpus of recent English text. The names *Zali* and *Steggall,* by contrast, would probably have a very low frequency, or even a zero frequency, even in a fairly large corpus of English.

The fact that *Zali* and *Stegall* are capitalized is a valuable clue that the tokens make up a name. However, the HMM formalism does not provide any systematic way to bias the tagging if a token is capitalized. The input to a HMM consists solely of the string of observed tokens. To take advantage of the capitalization attribute, what is needed here is some method which allows more than one clue to be taken into consideration; but there is no such general mechanism within HMMs.

To overcome this kind of shortcoming, **conditional random fields** (CRFs) were invented. CRFs first appear in the literature in 2001, and have been the subject of a considerable amount of research over the past decade.

1.1 An introduction to Conditional Random Fields

CRFs were first introduced by Lafferty et. al. [16]. Wallach [28] provides a good summary of the mathematics of this type of model.

The following section is a conceptual overview of CRFs with a focus on the ways in which CRFs can be applied to practical problems. This section may be skipped by the reader who is already familiar with CRFs.

1.1.1 Logistic Regression Models

By way of introduction, I will first briefly discuss logistic regression models. Logistic regression models share an important property with CRFs: both types of model are concerned with classifying individual units based on multiple attributes.

Following is an illustration of a logistic regression model. The example given here is from chapter 5 of Allison [2].

In a survey, 195 students were asked the question, "If you found a wallet in the street, what would you do?" The students could pick from three options:

Outcome	**Description**
0	keep both
1	keep the money, return the wallet
2	return both

In addition, the students were asked to classify themselves according the follow categories:

Dimension	**Description**	**Values**
1	Male	1: male 0: female
2	Business	1: enrolled in business school 0: not enrolled in business school
3	Punish	Variable describing whether student was physically punished by parents at various ages: 1: punished in elementary school, but not in middle or high school 2: punished in elementary and middle school, but not in high school 3: punished at all three levels
4	Explain	Response to question "When you were punished, did your parents generally explain why what you did was wrong?" 1: almost always 0: sometimes or never

Based on this input, it is possible to build a logistic regression model to predict the behavior of other individuals who were not included in the training set.

Note that the sequential ordering of the individuals is of no concern. If the training set or test set of students were shuffled, it would make no difference in the outcome.

1.1.2 Hidden Markov Models

Hidden Markov Models are so widely used in natural language processing that they hardly need any introduction. There are multiple good introductory discussions on HMMs, including those found in the computational linguistics textbooks of Manning and Schütze ch. 9 [19] (the illustration of the crazy soda pop machine is a particularly helpful one) and Jurafsky and Martin [12] (p. 241 ff.).

A typical application of HMMs is part-of-speech tagging. Given an input of natural language data, a HMM can be used to recover the most probable underlying part-of-speech category for each word:

VB	DT	NN	.		
Book	that	flight	.		
VBZ	DT	NN	VB	NN	?
Does	that	flight	serve	dinner	?

(Jurafsky and Martin [12] p. 299)

In classifying tokens, HMMs take two different factors into consideration. One set of probabilities has to do with the part-of-speech of a word in isolation. A token "book" could be either a noun or a verb, but if context is ignored, "book" might have a higher probability of being a noun, as estimated by the frequencies in a substantial tagged training corpus.

The other set of probabilities has to with the likelihood of sequences of categories. Perhaps the sequence "determiner - adjective - noun" is more probable than the sequence "determiner - determiner - determiner". Once again, these proba-

bilities are estimated on the basis of frequencies in a tagged training corpus.

1.1.3 Conditional Random Fields

Both HMMs and logistic regression models are concerned with classifying individuals, but the two kinds of model differ in at least two respects:

- Logistic regression models allow multiple attributes of the individuals to be taken into consideration. HMMs, by contrast, allow only the literal tokens themselves to be used as input.
- Logistic regression models do not take the sequential ordering of individuals into consideration. With HMMs, by contrast, the surrounding context of each token is one of the crucial factors.

CRFs can be thought of as combining properties of logistic regression models and HMMs. CRFs allow multiple attributes of individuals to be taken into consideration, as logistic regression models do. CRFs also consider the ordering of individual tokens, as HMMs do.

There are certain properties to CRFs which make them particularly convenient. For example, suppose the token *walked* appears in a text. This token has at least the following two attributes:

- The token has the literal form *walked*
- The token ends with the past tense suffix *-ed.*

These two attributes are not independent of each other. However, in training a CRF, the user does not need to take any particular action in connection with this dependency between attributes. This makes CRFs very easy to use.

CRFs are particularly well-suited to the problem of classifying dictionary tokens. As in the case of part-of-speech tagging, the tagging of dictionary elements involves the sequential ordering of the particular tokens. Dictionary tokens have multiple attributes which are useful for identifying token type (bold, italic, capitalization, presence of certain characters, etc.). HMMs have no straightforward way of taking these multiple attributes into consideration. CRFs, by contrast, are designed for exactly this sort of problem.

1.2 First case study: Lau

I turn now to two case studies involving the use of CRFs to tag dictionary entry tokens. The first case study involves a dictionary of Lau.

There are two languages named Lau, both in the Malayo-Polynesian family. The language under discussion here has the ISO 639-3 identifier **llu**; it is spoken on Malaita, an island in the Solomon Islands.

The dictionary studied here is the dictionary portion of *Grammar and Vocabulary of the Lau Language* (1920) by Walter G. Ivens. The dictionary was digitized by David Starner and was downloaded for the present study from the Project Gutenberg website.

The Lau dictionary is small in size, containing just 1365 entries. It was selected for the first case study because the entries are comparatively short and simple in their structure. As a crude measure of the complexity of the entries, there is a mean of 10.19 whitespace-separated tokens per entry in the Lau dictionary, compared with a mean of 65.45 tokens per entry in the Old English dictionary which will be considered in the second case study below.

Following are some typical entries from the Lau dictionary:

> *fou* 1. rock, stone; *si* fou, a rock. S. *hau*
>
> *fou* 2. v. i., to proclaim.

The dictionary allows for an entry to be followed by a subentry about a morphologically related word. This relatedness is indicated by indentation of the subentry:

> *nao* v. i., to lead; *nao tala,* lead the way; *eta inao,* to lead. S. *nao.*
>
> > *naofa* (*na*) n. eldest, first, *naofana mwela,* eldest child, *naofe mwela.*

In around 210 of the entries (15.4% of all of the entries), a portion of the entry consists of prose discussing grammatical aspects of the word. The second half of the following entry is an example of such prose.

> *gamoro* 1. pers. pron. dual 2. you two; used by itself as subj. or follows *igamoro.*

For the current study, the tokens in these prose discussion sections were tagged as a separate category from the English words in the definition field. Arguably, a definition is not the same kind of information as a discussion about grammatical aspects of the word.

1.2.1 Preparation of the data

The data posed few challenges in terms of preparation. The document contains only characters found in the ASCII set, which means that there were no character encoding concerns. Italics are indicated in the source document by putting underscores at the beginning and end of the italicized range; these markers were converted to ... tags in the initial file. Also, in the source document, an entry can be indented to indicate that it is a subentry of the preceding entry; the token INDENT was added to the beginning of these entries to encode this information.

1.2.2 Annotation of tokens with input tags

Each token in an entry has various attributes. Some of these are intrinsic to the text: for example, a token might be set in italics. Other attributes can be derived on the basis of external resources: for example, a token might be judged to be an English word if it appears in an external lexicon of English words.

Any approach to dictionary token tagging must somehow accomplish the following two tasks:

1. For attributes such as "italic" or "known English word," values must be determined and assigned to tokens.
2. There must be some sort of computation over those attribute values to assign a tag to each token.

Some authors have approached the problem of parsing a dictionary by writing a monolithic, dictionary-specific script (see e.g. Neff and Boguraev [20] p. 91 for a discussion and criticism of previous work). This sort of script tends to intermingle the two tasks listed above. For example, at the

point where the tagging logic needs to know whether a token is a known English word or not, it might look up the token in a hash whose keys are known English words. In this sort of architecture, the act of annotating a token with a value for the "known English word" attribute is implicit rather than explicit.

There are some drawbacks to this approach. First, multiple authors (e.g. Neff and Boguraev [20] p. 91) have commented on the need for reusable tools for dictionary text processing. A monolithic script is generally useful for only one dictionary, even though many of the subtasks, such as determining whether a token is a known English word, come up repeatedly across projects.

Second, this approach tends to obscure bugs because of its opaque nature. If there is a bug in the logic which determines whether a word is an English word, the bug might not be readily apparent from the output in which each token is tagged by type (headword, definition, etc.). Bugs can be more readily detected if the architecture makes all of the attribute values transparent to the developer.

The present work avoids the monolithic script approach, and instead adopts a modular approach to determining values of attributes. Annotating a dictionary with attributes is treated as the addition of tiers.

Stepping aside from Lau for a moment, consider a short entry from a Russian dictionary (figure 1). This entry might be represented in the initial text as follows:

```
<b>backpack</b> ['bækpæk] <i>n.</i> рюкза́к.
```

The tokens which make up this entry have various attributes which are inherently encoded within the text:

backpack	contains one or more boldface characters.
′bækpæk	is within brackets.
′bækpæk	contains IPA characters not normally found in English or Russian text.
n.	contains one or more italic characters.
n.	ends in a period.
рюкза́к.	contains one or more Cyrillic characters.
рюкза́к.	ends in a period.

Further attributes can be deduced based on external resources:

backpack	is found in an external dictionary of known English words.
n.	is found in a hand-prepared list of morphosyntactic abbreviations.

These attributes provide valuable clues to the structure of the entry. It is very convenient to represent this type of information as a multi-tiered annotation (figure 2).

backpack ['bækpæk] *n.* рюкза́к.

Figure 1: Russian (025)

For the present work, the dictionaries were represented as XML documents with one <token> element for each token, and a daughter element for each attribute tier. This approach worked perfectly well, although there was some noticeable slowness in the processing of lengthy documents. For future work, an implementation involving a relational database would probably be preferable in terms of speed and scalability.

A Perl package was written to allow convenient writing and reading of the multi-tiered representations. Various scripts were written which make use of this package; each script adds one or more tiers to the annotated dictionary. This general approach is pictured in figure 3.

Dictionaries are so diverse that nearly every dictionary will probably require some custom scripting. This observation has been made by others; for example, the system described by Neff and Boguraev [20] (p. 100) allows an "escape" to a general programming language (in their case, Prolog) to allow arbitrary processing.

On the other hand, there are certain tasks which are required for many dictionaries, such as the identification of English words based on an English word list. The approach described here allows reusable and custom tools to be conveniently used together.

1.2.3 Attributes of tokens in the Lau dictionary

Following is a list of the attributes which were used for the Lau dictionary. Some attributes, such as "italic", were obvious choices and were included from the start. Other attributes were determined through repeated runs of the CRF-based test; if it was observed that two tags were being confused, then an attempt was made to identify an attribute which could help the CRF to more reliably distinguish the two tags.

italic	The token contains at least one italic character
bold	Used here to indicate that the token is in a larger size (found only in headers indicating the start of the next initial letter)
parens	The token is partially or entirely within curved parentheses
brackets	The token is partially or entirely within square brackets
integer	The token contains at least one digit character
hw	The token appears to be the headword. (This is based on simple heuristics; for example, the first token in an entry is marked for this attribute unless the first token is the special INDENT symbol, and unless the entry is a single letter header to a section of the dictionary.)

	backpack	['bækpæk]	n.	рюкза́к.
boldface	1	0	0	0
italic	0	0	1	0
brackets	0	1	0	0
r-period	0	0	1	1
word-eng	1	0	0	0
chars-ipa	0	1	0	0
chars-cyrillic	0	0	0	1
pos-abbr	0	0	1	0

Figure 2: A multi-tiered representation of token attributes

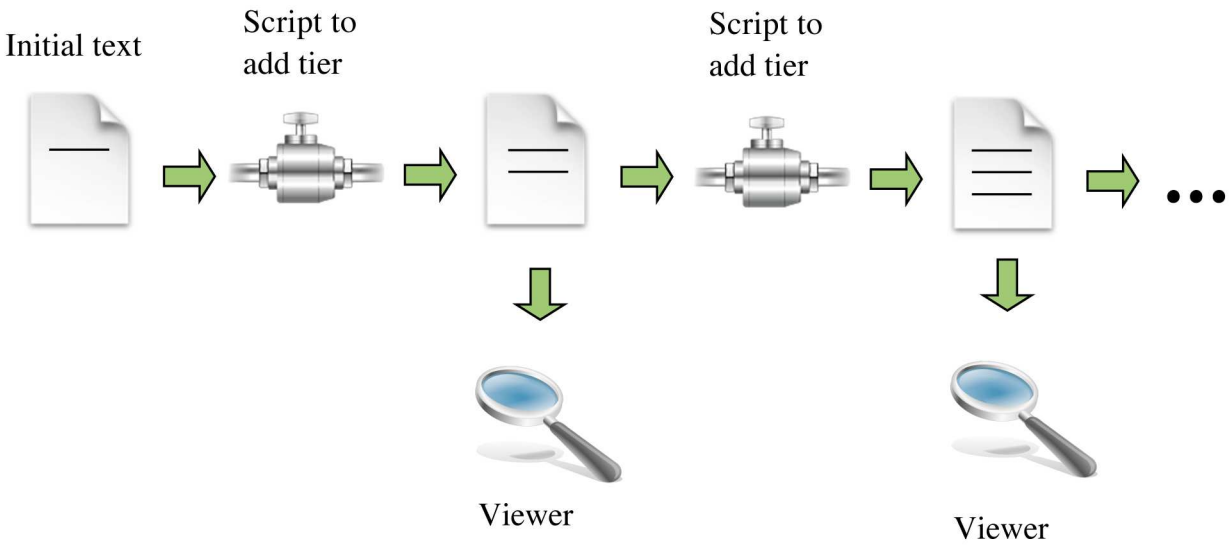


Figure 3: A modular approach to token attribute annotation

hw_int	The token appears to be an integer which disambiguates homograph headwords (e.g. *arai* 1, *arai* 2)
na_gu	The token appears to be a (*na*) or (*gu*) morphological marker (a category of token specific to this dictionary)
xref	The token appears to belong to a range of tokens making up a reference to another word
indent	The token is the special INDENT symbol which was inserted during acquisition
word_eng	The token is found in a dictionary of English words
pos	The token is found in a hand-collected list of abbreviations indicating part-of-speech or other morphosyntactic subcategory
pos_long	The token is found in a hand-collected list of unabbreviated words indicating morphosyntactic category (*personal article, prefix, negative,* etc.)
initial_caps	The initial letter of the token is a capital letter
r_comma	The final character of the token is a comma
r_period	The final character of the token is a period
r_semicolon	The final character of the token is a semicolon
prev_r_comma	The preceding token ends with a comma
prev_r_period	The preceding token ends with a period
prev_r_semicolon	The preceding token ends with a semicolon
q1	The token is found in the first quarter of the entry. For example, in an entry which is 20 tokens in length, the q1 attribute would be marked on tokens 1 through 5.
q2	The token is found in the second quarter of the entry
q3	The token is found in the third quarter of the entry
q4	The token is found in the fourth quarter of the entry

1.2.4 Tagging of data with output tags

Because the dictionary is relatively short, the entire dictionary was tagged for purposes of study.

Following is a sample entry from the Lau dictionary:

abalolo a banyan tree. S. *'apalolo.*

The tokens in this entry were coded as follows:

h d d d r r

(h = headword; d = English definition; r = cross-reference)

These tags are parallel to the part-of-speech tags for natural language text: the tag indicates the category of the token. Each token has exactly one tag.

1.2.5 The BIO strategy

The tags were processed further before the CRF model was trained. CRFs sometimes have trouble correctly identifying the beginning and end of a range. This problem can be helped by creating subtypes for each tag:

B = first tag in a sequence
I = tag within the interior of a sequence
O = last tag in a sequence

For example, consider the following imaginary sequence of tags:

a a a a a b b b c c c c c

These tags might be amplified as follows:

Ba Ia Ia Ia Oa Bb Ib Ob Bc Ic Ic Ic Oc

The present work uses only the B and I variants, but not O. The full BIO strategy, using all three variants (B, I, O) is appropriate in cases such as named entity recognition; a token either belongs to a name or else has no type at all. Using the O indicator in the final token in a sequence helps with the detection of the right edge. In the present case, however, the O designator is not needed, because the following token is of another non-null type.

Accordingly, the preceding Lau dictionary entry was automatically retagged by script to produce the following tag sequence:

h d d d r r → Bh Bd Id Id Br Ir

1.2.6 Tag set

The following token categories were identified. Note that these are the "output" categories. While a single token can be coded for any number of the preceding input attributes (italic, etc.), each token is tagged with exactly one output tag.

b	An integer distinguishing two homograph headwords
d	English definition of the Lau word, or an English translation of a sample Lau phrase

g	The markers (*na*) and (*gu*), which evidently have to do with the morphological category of the word
h	Headword
i	The special symbol INDENT, added during acquisition to encode the indentation of subentries
l	Section headers consisting of a single letter. For example, the header D appears before the section containing headwords beginning with *d*.
p	Abbreviations indicating part-of-speech or other morphosyntactic categories
r	Reference to another word, often in the closely related Sa'a language.
s	A sequence of words in Lau showing a sample use of the headword in context
x	An English prose discussion about the word, typically on the grammatical properties of the word, to be distinguished from the English definition of the Lau word

Two strategies were used to validate the tagging. First, the viewer program includes functionality to generate a separate HTML page for each tag. For example, there is a "b" page containing all and only the contents of the "b" fields. An error in the tagging of this field is often immediately apparent on visual inspection.

Second, a CRF model was trained over the entire data set, and then the entire data set was tagged using the CRF model. Obviously, using the same data set for both training and testing is unacceptable for evaluation purposes, but it is very useful as a way of detecting errors in the tags assigned by hand. In cases where there was a mismatch between the hand-assigned tag and the CRF-assigned tag, it was commonly the case that the error was on the human side.

1.2.7 Mallet

For the training and evaluation of the CRFs, the Mallet software package was used.

Mallet ("MAchine Learning for LanguagE Toolkit") is described on the Mallet project homepage as "a Java-based package for statistical natural language processing, document classification, clustering, topic modeling, information extraction, and other machine learning applications to text." Mallet was developed by Andrew McCallum with the assistance of the others; it contains support for CRFs, among other kinds of models.

Following is a part of the input to Mallet which was used to train the CRFs for the Lau dictionary. This sample corresponds to the entry cited above.

```
abalolo hw italic q1 Bh
a q2 word_eng Bd
banyan q3 word_eng d
tree.  q4 r_period word_eng d
S. prev_r_period q4 r_period xref Br
'apalolo.  italic q4 r_period xref r
```

Each row corresponds to one token in the dictionary. The first symbol in each row is the literal token itself. The last symbol is the tag. The other symbols on each line indicate token attributes. If a token is set in italics, then the symbol "italic" appears on the corresponding row; but the fact that a token is not italic is not explicitly indicated.

The CRF treats the literal token itself (abalolo, a, banyan, tree...) as one of the attributes. If the training set contains multiple tokens *adj.* which are hand-tagged with the "part of speech" type, there is a good chance that the CRF will tag further tokens of *adj.* as "part of speech" even if there are no other attributes added to help the CRF recognize that category of token.

The test data is formatted similarly to the training data, except that the category is omitted from the end of the line.

1.2.8 Method

After the data was annotated and tagged, it was divided into a training set and a test set. The two sets contain a nearly equal number of entries.

The division was accomplished by alternating entries: entry 1 is assigned to the training set, entry 2 is assigned to the test set, entry 3 is assigned to the training set, and so on. This approach helps avoid introducing any bias into the training or test sets which might exist in different regions of the dictionary. (In the second case study, to be discussed below, this is a real concern, because different portions of the dictionary were written by different authors who did not always follow exactly the same conventions.)

A question of practical concern is this: how much data must be tagged by hand to achieve a reasonable level of accuracy? To help answer this question, the training set was further processed into ten training files: one containing around 10% of the total training data, one containing around 20%, and so on, up to a tenth file containing 100% of the training data.

Using the Mallet program, ten CRF models were trained, one from each of the ten training data sets. Then the test data was tagged ten times, once with each of the ten CRF models.

1.2.9 Scoring

The data was scored by comparing the hand-assigned tags with the Mallet-assigned tags. The score is a simple percentage of the tokens which were tagged correctly by Mallet. The scores for the ten models are graphed in figure 4.

Following are the raw numbers:

Test number	**Number of tokens in training set**	**Accuracy**
1	645	86.43
2	1408	92.22
3	2386	94.16
4	3124	95.33
5	3873	95.23
6	4766	95.59
7	5456	95.36
8	6170	95.74
9	6952	95.74
10	7729	95.63

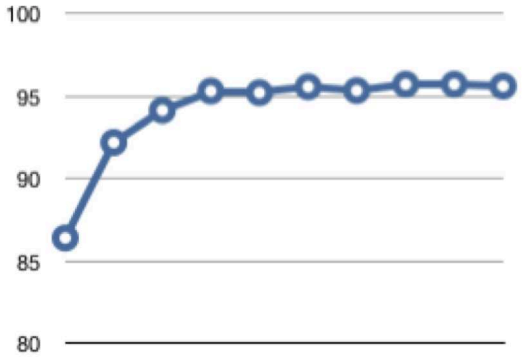


Figure 4: Token tagging accuracy rates for the Lau dictionary

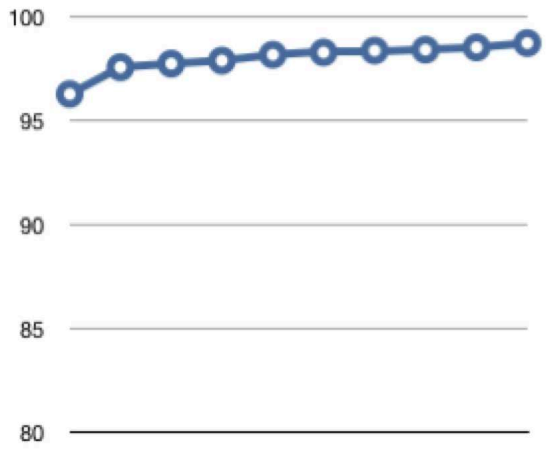


Figure 5: Token tagging accuracy rates for the Lau dictionary when entries containing "x" tag are omitted from training and test sets

The finding is that there is little improvement beyond the fourth test; the accuracy stubbornly remains at just above 95%, even when the amount of training data is more than doubled.

Note that the accuracy does not improve monotonically as additional training data is added. For example, the test where the training set contains 100% of the tagged training data yielded a slightly lower accuracy than the test with 90%. This is probably because the additional data happened to include entries of uncommon types which conflict with the form of more common types, creating greater uncertainty in the model.

The specific tagging errors were studied. By far, the most frequent confusion is between **d** (the English definition) and **x** (English prose discussing the grammatical matters). This is not especially surprising, since both fields consist mainly of English words. The two are distinguished partly by the choice of words, and partly by position within the entry. These two clues are of limited helpfulness, however. There is overlap in the words found in two types of fields; and both fields can appear in multiple positions within the entry.

How well would the CRF do if the problematic distinction between **d** and **x** did not exist in the dictionary? Out of curiosity, a second experiment was run which followed the same method just described, except that all of the entries containing the tokens with the problematic **x** tag were removed from the data prior to splitting the data into training and test sets (obviously, this would not be permissible in a real-world case, but it is useful as a way of studying how much of the error rate can be attributed to the presence of the **x** fields). Following is a comparison of the dictionary before and after this artificial removal of entries:

	Number of entries	**Number of tokens**
Full dictionary	1384	15,483
"x" entries removed	1174	10,724

The results are graphed in figure 5. Obviously, removing the confusability of the entries containing **x** tags makes a considerable difference. The accuracy on the full data set (where the data contains the **x** tags, and where 100% of the data is used) is 95.63%. When the entries containing **x** tags are removed, the accuracy goes up to 98.71%.

1.3 Second case study: Old English

For the second study, a much larger and far more complex dictionary was chosen: *An Anglo-Saxon Dictionary* by Joseph Bosworth and T. Northcote Toller (main volume published in 1898; supplementary volume published in 1921). The work has been the definitive lexical reference on Old English for over a century, although it is gradually being superseded by the *Dictionary of Old English* as fascicles of the latter work are being published over a period of decades.

The dictionary consists of a main volume and a later supplement. The main volume is 1302 pages long, and the supplement volume is 768 pages long. The pages are large and are densely set in two-column format in small type. The dictionary as a whole contains upward of 60,000 entries. An exact count of entries is difficult to determine, because some of the entries in the supplement volume are corrections on entries in the main volume, rather than free-standing entries on words not included in the main volume.

An Anglo-Saxon Dictionary has a complex history. Joseph Bosworth published an earlier work, *Dictionary of the Anglo-Saxon Language,* in 1838. He undertook a major revision of the work, but did not complete it; at the time of his death in 1876, the revision existed in incomplete manuscript form. Thomas Northcote Toller took on the project and completed the main volume.

Due to this history, there are inconsistencies in the format of the entries in the main volume. There are some lengthy ranges of entries which fairly clearly show the mark of a different author.

A sample entry from Bosworth/Toller is pictured in figure 6. The sample entry has the following structure:

Headword
Alternate spellings or forms of the headword
Morphosyntactic abbreviations
Modern English definition
Latin definition
The special marker :— which introduces the citation section
A number of citations, each containing:
  A quotation in Old English
  A translation into English or Latin
  A list of text citations in which the phrase in question appears (e.g. Exon 40b)

This entry is very typical in terms of its structure. A reasonably-constructed grammar which accepts this particular entry would serve to accept a fairly large percentage of the entries in the dictionary. A few more refinements would expand this coverage. Some entries allow numbered subentries, where each subentry includes the English/Latin definition and the citation section. Also, an etymology field and/or a cross-reference to another entry is permitted to appear at the end of the entry.

The pages of the dictionary are information-dense. The single entry in figure 6 comprises just 4% of the text on one page, based on a count of individual characters. Some entries are quite long, stretching on for a page or more.

1.3.1 Preparation of the data

The dictionary had already been digitized through OCR and hand-corrected by volunteers. For the present study, the data was converted to UTF-8.

A sample of 13 pages was extracted from the main volume (pages 100, 200, 300, etc.), comprising 306 entries. Due to the density of the information on the page, the sample comprises 20,333 tokens—a substantial number for hand-tagging.

Pages from the supplement volume were not included in the study, because they would clearly require a different model; the entries in the supplement are often corrections on entries in the main volume, and often consist of instructions as to sections of text which should be inserted into or deleted from existing entries.

Using the multi-tiered approach described earlier, the sample was annotated with the following attributes:

italic	The token contains at least one italic character
bold	The token contains at least one bold character
parens	The token is partially or entirely within curved parentheses

firen-full, fyren-full, -ful; *adj. Sinful;* fācfnŏrōsus, scĕlestus:—Swâ firenfulle heora aldorþægn unreordadon *thus the sinful addressed their principal chief,* Cd. 214; Th. 268, 34; Sat. 65. Gif đū wylt đa firenfullan fyllan mid deáþe *if thou wilt fell the wicked with death,* Ps. Th. 138, 16. Firenfulra *of the wicked,* Exon. 40b; Th. 135, 30; Gū. 532: Ps. Th. 81, 4: 124, 3.

Figure 6: A sample entry from Bosworth/Toller (025)

brackets	The token is partially or entirely within square brackets
integer	The token contains at least one digit character
non_ascii	The token contains at least one character not found in the ASCII set, such as á, ð, or ŭ. (Crudely, this attribute distinguishes one group of token types, such as modern English definitions and subentry numbers, from a complementary group, including headwords, Latin definitions and citation translations, etc.)
chars_germanic	The token contains characters which are specific to Old English and the other early Germanic languages, such as ð and þ.
chars_greek	The token contains Greek letters, such as Δ, ε, ξ, etc.
chars_latin	The token contains characters which are found in this dictionary only within Latin words, such as ā, ŏ, ū.
initial_caps	After leading punctuation is stripped off, the initial character of the token is a capital letter
initial_hyphen	The initial character of the token is a hyphen
final_hyphen	The final character of the token is a hyphen
l_bracket	There is a left square bracket at the beginning of the token
r_bracket	There is a right square bracket at the end of the token
r_comma	There is a comma at the end of the token
r_period	There is a period at the end of the token
r_semicolon	There is a semicolon at the end of the token
short	The token is "short" (including punctuation, the token is shorter than 5 characters)
word_ang	The token appears to be an Old English word. More specifically, the token appears in the comprehensive Toronto *Dictionary of English* corpus of Old English text. String matching in this case involves some complications, because the Old English words in the dictionary are written with diacritics which are not included in the DOE corpus.

word_eng	The token is found in a lexicon of Modern English words
word_lat	The token is found in a list of known Latin words. (The list was prepared for the current project from a downloaded copy of the Vulgate. Many Old English texts concern ecclesiastical matters, which means that there is a decent overlap between the Vulgate and the dictionary not only for high-frequency function words, but also for lower-frequency content words.)
latin_suffix	The token ends in a common Latin suffix such as *-orum* or *-ibus*. The list of suffixes was prepared largely by hand, based on frequency counts of terminal character sequences in the Vulgate.
abbr_morph	The token is found in a list of known morphosyntactic abbreviations which are specific to this dictionary.
abbr_table	The token is found in the table of abbreviations of Old English texts. This table is found in the introductory pages of the main volume. For example, Beo. Kmbl. refers to an edition of Beowulf edited by John M. Kemble.
roman_num	The token is one of the following strings: I II III IV V VI VII VIII IX X
v	The token is the keyword "v" (= "see also"), or is the token immediately following a "v" token.
etymology	Based on a complex set of heuristics, the token appears to be within an etymology field. The etymology tends to appear near the end of the entry; it is always enclosed entirely in square brackets; and it generally contains certain distinctive abbreviations not found elsewhere in the entry, such as O. H. Ger. or O. Icel.
der	The token is the keyword DER, or is the token immediately following DER. DER indicates that the following word is related to the headword through some kind of derivational morphology or compounding.
div_sym	The token is the special divider symbol :—, which separates a definition field from a citation section.
div_left	Within a subsection of the entry, the token appears to the left of the :— symbol.
div_right	Within a subsection of the entry, the token appears to the right of the :— symbol.
q1	The token is found in the first quarter of the entry
q2	The token is found in the second quarter of the entry
q3	The token is found in the third quarter of the entry
q4	The token is found in the fourth quarter of the entry

The data were tagged by hand. Following are the token categories which were recognized:

a	An alternate form of the headword, listed immediately following the headword
c	Citation information (e.g. Exon 98 a; Th. 368, 33; Seel. 34)
d	Cross-reference to a word related to the headword by derivational morphology or by compounding
e	Modern English definition of the headword
g	Greek translation of a cited Old English text
h	Headword
l	Latin translation of the headword
m	Morphosyntactic abbreviation section
n	Subsection numbers or letters (I, 1, a, etc.)
q	Quoted Old English text within a citation
r	Bracketed comment after the headword; when the headword is a compound, this comment often contains the definition of one or more of the members of the compound
t	Modern English translation of a cited Old English text
u	Latin translation of a cited Old English text
v	Cross-reference to another headword
w	The special :— symbol
y	Etymology

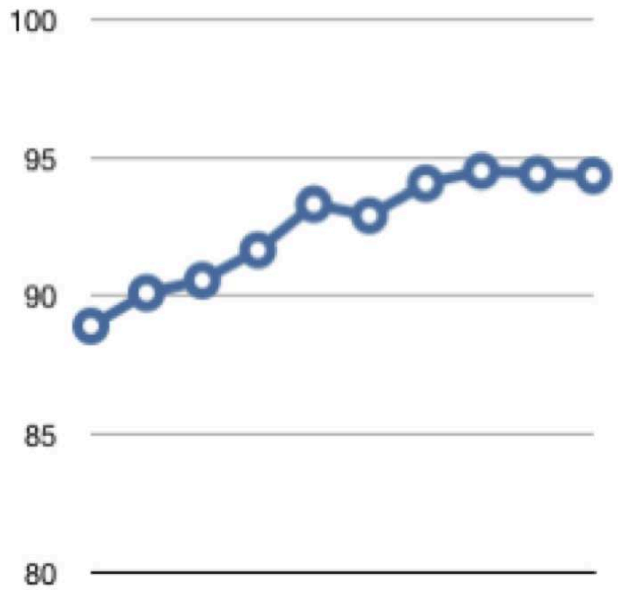


Figure 7: Token tagging accuracy rates for the Old English dictionary

O	The token is of an irregular type and was left unclassified. By convention, O is used in Mallet to indicate unclassified tokens.

As with Lau, the tagging was validated in part by using the viewer tool to separate out ranges belonging to a single tag (for example, all of the Latin definitions appear together on one web page); errors were often immediately apparent. Also, consistency checks were repeatedly run; a CRF data was trained on all of the data, and all of the data was tagged using that model. As with Lau, it was very often the case that discrepancies between the hand-assigned tags and the CRF-assigned tags were due to errors in the hand tagging. This sort of consistency check was found to be of great practical value.

1.3.2 Method

The evaluation method was identical to that used in the study of the Lau dictionary. The data was divided into a training set and a test set of roughly equal size. Then 10 training files were produced, respectively comprising approximately 10%, 20%, 30% etc. of the data. A CRF model was trained on each training file using Mallet, and the test set was evaluated against each model.

1.3.3 Results

The results of the ten test runs are graphed in figure 7. As with Lau, the numbers are the percentages of tokens where Mallet assigned the correct tag.

Test number	Number of tokens in training set	Accuracy
1	1206	88.92%
2	2257	90.11%
3	3048	90.56%
4	4123	91.65%
5	5727	93.31%
6	6613	92.90%
7	7423	94.06%
8	7927	94.50%
9	8720	94.39%
10	10376	94.34%

As expected, the numbers are somewhat lower than those for Lau, corresponding to the substantially greater complexity of the Old English dictionary.

As with Lau, the chart reaches a plateau, and additional data does not noticeably improve accuracy. The Lau data set reached its plateau around test 4, with around 3124 tokens; but the Old English data reaches its plateau later, around test 7, with 7423 tokens. This difference can be attributed to the substantially greater complexity of the entries in the Old English dictionary.

The most common confusion was that where **t** (Modern English translation of a cited Old English passage) was mistakenly tagged by Mallet as **e** (Modern English definition of the headword). In test 10, where the model was trained on 100% of the data, this particular confusion accounted for 1.12% of all tokens in the test set. This confusion is unsurprising, since both fields consist of modern English words set in italics.

The special tag **O** (unclassified token) was involved in many of the high-running confusion types, either as the hand-tagged value or as the Mallet-tagged value of the confusion pair. For example, the second most frequent confusion, accounting for 0.89% of the tokens, is the case where a token hand-tagged as **y** (etymology) was mistagged by Mallet as **O**. This is unsurprising, since both **y** and **O** fields contain tokens of multifarious types.

1.4 Discussion

With both the Lau and Old English dictionaries, it was observed that adding additional training data is helpful only up to a point; the trend in improvement reaches a plateau, and even doubling the amount of training data might result in improvement of only tenths of a percent. The specific findings can be summarized as follows:

	Lau	Old English
Mean number of tokens per entry	10.19	65.45
Approximate number of training tokens where a plateau was reached	3124	7423
Percentage accuracy at the point where a plateau was reached	95.33%	94.06%

These two examples provide a rough guide for the use of CRFs in dictionary token tagging. If one were coding a dictionary which has a mean of 25 tokens per entry, one could use the preceding chart to make some rough predictions regarding the number of tokens which one should expect to

have to tag, and the level of accuracy which can be expected.

For some applications, an accuracy of 94 $\sim$ 96% might be unacceptably low. CRFs are so convenient and easy to use that a reasonable approach in such cases is to use a CRF to do an initial pass of tagging, and then to correct the result (Fomin and Toner [7] p. 84 discuss a similar case of editorial correction after automated markup). One approach is to make corrections by hand. Another approach is to use some kind of a script to identify possible errors. From the perspective of the multi-tiered approach discussed earlier, the tags output by the CRF can be treated as just another tier; the error detection scripts can therefore readily have access to the full range of annotations.

There might also be cases where an accuracy of 94 $\sim$ 96% is acceptable. As noted, not all types of error are observed to be equally likely; there are certain token types which tend to be confused in specific ways. In the case of the Lau dictionary, for example, the **d** tag (English definition) and the **x** tag (English prose description of some aspect of the word) were observed to be confusable. There might be some applications where the distinction between these two fields is unimportant. How much post-automation correction is required will depend on the specifics of the situation.

2. SURVEY OF DICTIONARIES

We now turn away from a specifically CRF-based approach to discuss the findings of a survey of 100 bilingual dictionaries.

Researchers who have worked on the processing of dictionary text have long expressed a desire for a general set of tools (e.g. Neff and Boguraev [20] p. 91). Each dictionary has its own idiosyncrasies, which means that every dictionary processing project will probably require some custom scripting and/or custom categories of markup. For example, the TEI tag set includes tags specific for dictionary markup, and it was designed to be comprehensive; but at least three authors have had to create extensions to the TEI tag set to accommodate dictionary-specific needs (Nyhan [22] p. 8, handling Irish consonant mutation; Fomin and Toner [7] p. 85-88 distinguishing active and passive forms of verbs; Kang [14] handling morphologically irregular forms, vowel length, and conventional points for hyphenation breaks, among other items).

On the other hand, when any substantial collection of bilingual dictionaries is viewed as a population, there are certain features which come up frequently across dictionaries. Ideally, a set of tools will be architected in a way which allows custom tools to be conveniently used when needed, but will provide ready-made solutions for the frequently recurring tasks. The aim in this section is to make a systematic inventory of these frequently recurring tasks, so that tools can by built in a planned and considered way.

A typical sort of finding from the dictionary survey is that 25 out of 100 dictionaries allow parentheses in the headword to indicate optionality, as in **colo(u)r**. This is not an especially profound finding, but it is a useful one as a part of a checklist of dictionary features which must be taken into consideration in the creation of a general set of tools. A comparison can be made here with the work of land surveyors: the focus is not on the production of some theoretically deep result, but rather is a matter of systematically creating a comprehensive overview for practical use.

2.0.1 Overview of the survey

100 bilingual dictionaries were surveyed. This number conveniently allows counts and percentages to be used interchangeably. The two halves of a bidirectional dictionary (e.g. German $\rightarrow$ English, English $\rightarrow$ German) are counted and coded as separate dictionaries, because there are usually substantial differences in the format of the entries (this will be discussed further below). If each bidirectional dictionary is counted just once, then the count of surveyed dictionaries is 70.

A survey was prepared consisting of 57 attributes of bilingual dictionaries. Following is a short sampling of the survey items:

> Is there any boldface type present in the dictionary entries? (Y/N)
>
> Does the dictionary include pronunciations? (Y/N)
>
> If pronunciations are present, are they written in the International Phonetic Alphabet? (Y/N)

The list was prepared by doing an informal initial survey of the 100 dictionaries; the intent was to come up with a list which would account for as much of the variation among dictionaries as possible. Of course, there is some irreducible subjectivity to the selection of attributes, and the list of attributes is not exhaustive. However, if another researcher were to independently prepare such a list, there would almost certainly be a substantial overlap with the list used here. It is difficult to miss the fact that some dictionaries include pronunciations and that others do not. The major features of the terrain are too prominent to be easily missed.

2.0.2 Selection of the dictionaries

The selection of dictionaries was informal. An effort was made to cover a wide variety of bilingual dictionaries. In particular, an effort was made to give fair representation to all of the following categories:

- Both European and non-European languages
- Languages of major international commercial and political significance (Russian, Japanese, French, Arabic) as well as lesser-taught languages (Luganda, Indonesian, Welsh, Chickasaw, Tibetan)
- Dictionaries published within the past few decades, and dictionaries published many decades ago
- Dictionaries whose entries are fairly simple in structure, and those which are very complex
- Large references which aim for comprehensiveness and completeness, and short references which aim for compactness and convenience

While the project is primarily concerned with general-purpose dictionaries of modern languages, a few dictionaries of the following types are also represented in the sample:

- Dictionaries whose selection of words is limited to a specialized domain (e.g. computer terminology)
- Dictionaries of idioms (as long as the entry format resembles a conventional dictionary; a book discussing idioms in prose form would be considered out of scope)

contextual [kɔn′tekstjuəl, -tʃwəl] contextueel
contiguity [kɔnti′gjuiti] aangrenzing, nabijheid

Figure 8: English → Dutch (022a)

′aanbevelen (beval ′aan, h. ′aanbevolen) **I** *overg* recommend, commend; *wij houden ons aanbevolen voor…* we solicit the favour of… [your orders]; **II** *wederk*: *zich* ~ recommend oneself

be′ëdigen (beëdigde, h. beëdigd) *overg* 1 (iem.) swear in [a functionary]; administer the oath to [the witnesses]; 2 (iets) swear to, confirm on oath
be′ëdiging *v* (-en) 1 swearing in [of a functionary]; 2 administration of the oath [to witnesses]; 3 confirmation on oath

Figure 9: Dutch → English (022b). The two halves of the bidirectional dictionary do not have the same entry structure.

- Dictionaries of premodern languages (Latin, etc.)

The following types of dictionaries are treated as out of scope and were not included in the sample:

- Monolingual dictionaries
- Travelers' phrase books. It should be noted, however, that there is not a well-defined line separating small dictionaries from glossaries within phrase books.
- Pictorial dictionaries
- Dictionaries of sign languages
- References whose primary function is a cataloging of orthographic characters together with prose discussion of the characters (e.g. Henshall, Kenneth G. 1988. *A Guide to Remembering Japanese Characters.* This text is organized by character, and provides sample uses of each character together with English translations, which makes the text resemble a dictionary in some respects; but each entry consists largely of prose description of the history of the character, together with helpful mnemonics for memorizing the character.)

2.0.3 Bidirectional dictionaries

As previously noted, the two halves of a bidirectional dictionary are treated as separate dictionaries. It was observed as a general pattern that there are substantial differences between the two halves of a single bidirectional dictionary.

For example, consider an English ↔ Dutch dictionary. Figure 8 contains some entries from the English → Dutch dictionary; figure 9 contains some entries from the Dutch → English section.

These entries illustrate some of the differences between the two sections. The English → Dutch section contains pronunciation fields, but the Dutch → English section does not. The Dutch → English section contains accent marks in the headwords to show the word stress patterns (**′aanbevelen, be′ëdigen**). The Dutch → English entries also contain fields indicating the inflected forms of the headword (beval′aan, h. ′aanbevolen). Neither of these features are found in the English → Dutch section.

This short set of differences can be summarized as follows:

	Dutch → English	**English → Dutch**
Contains pronunciations?	No	Yes
Stress marks in headwords?	Yes	No
Includes inflected forms?	Yes	No

Differences of this sort were found to be common in bidirectional dictionaries. For most practical applications, it would probably not be very useful to attempt to model both halves of the dictionary with a single model. Accordingly, for the present study, a separate file was coded for each half of each bidirectional dictionary.

2.0.4 Method

From each dictionary in the sample set, copies of the following pages were collected:

- First page of main body
- Page 50
- Page 100
- Last page of main body

In some cases, there was a problem which prevented these guidelines from being followed exactly. For example, it might happen that page 50 of a particular dictionary is nearly blank because it is the last page for the entries beginning with a particular letter. In cases such as these, a suitable substitution was made.

In the case of bidirectional dictionaries, the sample of four pages was collected from each of the halves of the dictionary.

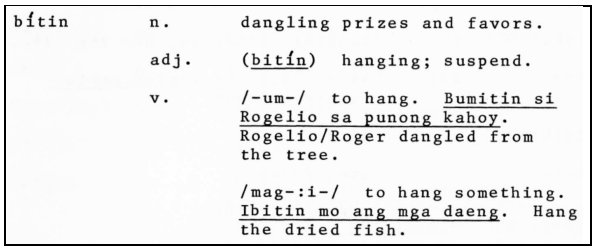
The small size of the sample (four pages per dictionary) imposes some limitations. There are some features which may occur only rarely within a dictionary and might not happen to occur within the sample. This almost certainly has led to some undercounting. For example, there were dictionaries whose introductory section included a table of domain indicators such as *mil., med., chem.,* or *mus.,* but where none of these subject area indicators were observed anywhere in the four-page sample.

On the other hand, many of the attributes involve items which occur so frequently that the small size of the sample does not pose any difficulty (only a trivial sample is required to determine whether the headwords are set in boldface, for example). It would obviously be impractical to exhaustively study every page of every dictionary. The survey has revealed some very clear general tendencies. A substantially larger sample of pages from each dictionary would probably contribute only a small amount of improved accuracy.

2.1 Findings

2.1.1 Layout as a structure indicator

Among the surveyed dictionaries, 28% make use of layout to encode some aspect of the structure of the entry.

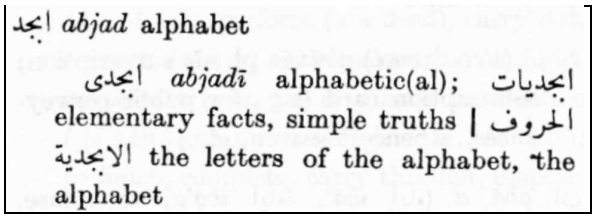

bítin n. dangling prizes and favors.

adj. (bitín) hanging; suspend.

v. /-um-/ to hang. Bumitin si Rogelio sa punong kahoy. Rogelio/Roger dangled from the tree.

/mag-:i-/ to hang something. Ibitin mo ang mga daeng. Hang the dried fish.

Figure 10: Tagalog (016). Layout as a structure indicator.


ابجد *abjad* alphabet

ابجدى *abjadī* alphabetic(al); ابجديات elementary facts, simple truths | الحروف الابجدية the letters of the alphabet, the alphabet

Figure 11: Arabic (031). Narrower margins as a structure indicator.

Consider the Tagalog dictionary entry in figure 10. In terms of its abstract structure, the entry is of a common type: there is a first-order branching by part of speech, and a second-order branching within the part of speech. What is unusual about this dictionary is the way in which this structure is presented. Most dictionaries would format the entry as a continuous flow of text with the structure indicated by numbering (**1. 2. 3. a. b.**), but the Tagalog dictionary instead indicates the structure of the entry through the graphical positioning of elements.

A fairly common use of layout is a region of text with narrower margins beneath the main entry. This is found in 10% of the surveyed dictionaries. Figures 11 and 12 contain examples.

Both the Indonesian and the Arabic dictionaries place morphologically derived words in an indented section after the main entry. The indented section in the Arabic dictionary is a continuous flow of text, but the Indonesian example makes further use of layout to indicate structure by starting a new line for each derived word.

The Bengali entry in figure 13 also uses indentation, but with different semantics from that in the Arabic and Indonesian examples.

Instead of morphologically derived forms, the information in the indented section is of two types: a special morphological form ("*Negative:* nai."), and examples of the use of the entry word ("mač bhalo ače : Fish is good."). As with the Indonesian example, newlines are used within the indented

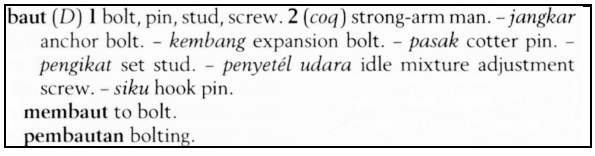

baut (*D*) **1** bolt, pin, stud, screw. **2** (*coq*) strong-arm man. – *jangkar* anchor bolt. – *kembang* expansion bolt. – *pasak* cotter pin. – *pengikat* set stud. – *penyetél udara* idle mixture adjustment screw. – *siku* hook pin.
membaut to bolt.
pembautan bolting.

Figure 12: Indonesian (009). Narrower margins as a structure indicator, with each subentry on a new line.

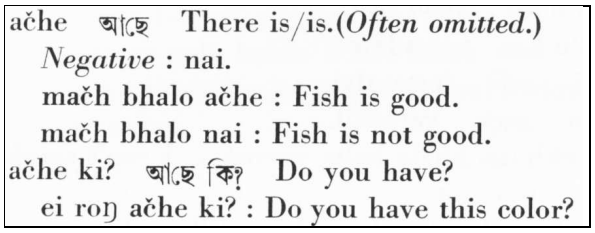

ače আছে There is/is.(*Often omitted.*)
Negative : nai.
mač bhalo ače : Fish is good.
mač bhalo nai : Fish is not good.
ače ki? আছে কি? Do you have?
ei roŋ ače ki? : Do you have this color?

Figure 13: Bengali (011). A different use of narrower margins.

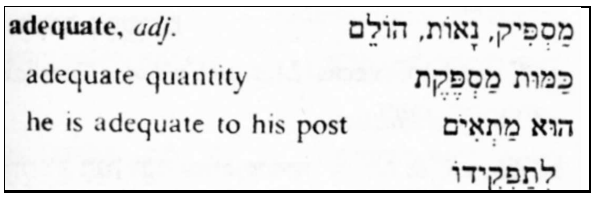

adequate, *adj.* מַסְפִּיק, נָאוֹת, הוֹלֵם
adequate quantity כַּמּוּת מַסְפֶּקֶת
he is adequate to his post הוּא מַתְאִים לְתַפְקִידוֹ

Figure 14: Hebrew (019). Columns as a structure indicator.

section to encode further divisions. However, in the Bengali example, not all lines within the indented section contain the same type of data. The presence of indentation narrows down the set of possible types of information, but some further heuristic would be needed to complete the classification of each line.

A less commonly used layout element is columns. In the Hebrew dictionary in figure 14, columns are used to separate the elements in the headword language from the elements in the translation language.

2.1.2 *Encoding significant layout*

Layout is a valuable clue to the structure of the text. It has been noted in the literature that significant layout information must somehow be encoded and preserved during the acquisition stage (e.g. Kammerer [13] p. 22).

One strategy for encoding this information is to embed presentation-level tags in the initial text (e.g. <indented> ... </indented>); each token within the denoted range can be considered to have a value of 1 for an "indented" property. In section 1, a different approach was used in the case of the Lau dictionary; an INDENT symbol was inserted as a separate token, ensuring that this information remains available after the tagging process.

2.1.3 *Graphical delimiters*

22% of the dictionaries in the survey made use of at least one special graphical symbol as a non-typographic structure indicator. This count excludes the most common punctuation characters: brackets, parentheses, periods, commas, colons, and semicolons.

The Tibetan entry in figure 15 contains an example of the use of the ¶ symbol as a non-typographic structure indicator.

Following is a count of the observed symbols used to indicate something about the structure of the entry:

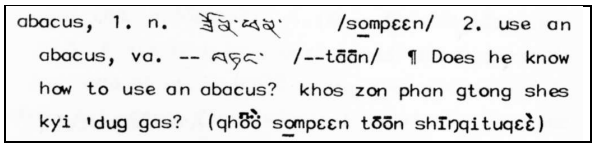


Figure 15: Tibetan (024). Use of ¶ as a graphical delimiter.

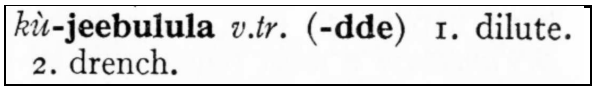


Figure 16: Luganda (036). Text attributes change within a word. This entry is alphabetized under J, not under K.

Count	Type
5	double pipe || (In one dictionary, the pipe to the left is much thicker)
4	slash /
3	paragraph symbol ¶
2	lozenge-like symbol
1	pipe |
1	em-dash —
1	colon plus em-dash: :—
1	circle •
1	number sign §
1	hand ☞
1	right triangle ▶
1	multiple: lozenge ◆ and right triangle ▶

2.1.4 Text attributes

Most, but not all, dictionaries use text attributes such as boldface or italics as typographic structure indicators. Following are the observed frequencies of the use of these attributes:

85%	Bold
85%	Italics
23%	Superscript
3%	Underline

It was also observed that one dictionary uses a smaller typeface as a typographic structure indicator.

2.1.5 Switching text attributes within one word

Some dictionaries permit a change in value within a single token for text attributes such as bold or italic (figures 16, 17, and 18).

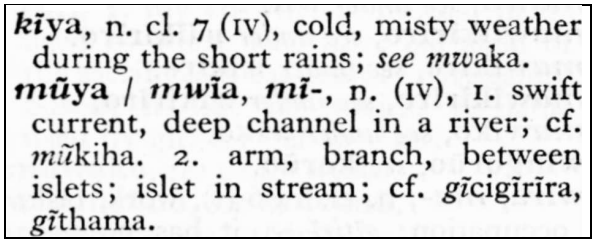


Figure 17: Kikuyu (037). Text attributes change within a word.

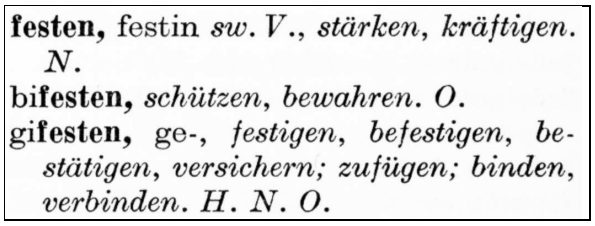


Figure 18: Old High German (029). Text attributes change within a word.

Note the way that the entries are alphabetized in the Old High German example (figure 18). One approach to parsing this dictionary would be to treat bi**festen** as a subentry to **festen**, but there are also cases in this dictionary where an entry with a non-bold prefix can stand on its own as its own entry, without a preceding unprefixed main entry. An entry of that type is alphabetized as if the prefix were dropped.

This kind of token-internal attribute change was not found in the Lau or Old English dictionaries considered in the case studies above.

Some approaches, such as the CRF-based approach considered in section 1, require that attributes be encoded at the token level rather than at the character level. A reasonable approach is to define the "bold" attribute to mean that the token contains at least one bold character. Thus, a token such as *kù*-**jeebulula** would have the value TRUE for both the "italic" and "bold" attributes.

Of course, some applications require the character-level attributes to be preserved; this would be true in a web-based dictionary search system, where the correct character attributes are required for presentation purposes. In a multi-tiered form of annotation, a tier can contain the token together with markup tags to encode the character-level attributes.

2.1.6 Entry-internal numbering

41% of the dictionaries contain some kind of numbering of subsections within the entry. 20 of these 41 dictionaries indicate numberings in bold; 17 of the 41 dictionaries have a period following the number. One dictionary sets the subentry numbers in italics.

In two dictionaries, it was observed that the majority of subentry numbers are followed by a period, but it was observed that a comma is sometimes mistakenly used instead of a period.

Following is a breakdown of the numbering methods:

Count	Type
22	1. 2. 3.
8	1 2 3
2	① ② ③
2	1 a b 2 a b
1	1) 2) 3)
1	(1) (2) (3)
1	(a) (b) (c)
1	I II III
1	I 1. 2. II 1. 2.
1	1. a. b. 2. a. b.
1	I. A. 1. a. (some entries use Greek letters instead of lower-case Roman letters, with no obvious difference in meaning)

bad [bæd] *n.* **1.** (*evil*) дурно́е, плохо́е; ху́до; **go the ~** разор|я́ться, -и́ться; сби́ться (*pf.*) с пути́ и́стинного. **2.** (*loss*): **I was £5 to the ~** я понёс убы́ток в пять фу́нтов.
adj. **1.** плохо́й, дурно́й, сквёрный; **not ~!** непло́хо!; **things went from ~ to worse** дела́ шли всё ху́же и ху́же; **too ~!** о́чень жаль!; **it is too ~ of him** э́то о́чень некраси́во с его́ стороны́; **a ~ light** (*to read in*) сла́бый свет. **2.** (*morally bad*) плохо́й, дурно́й; **it is ~ to steal** ворова́ть (*impf.*) ду́рно/пло́хо; **lead a ~ life** вести́ (*det.*) непутёвую/беспу́тную жизнь; **a ~ name** дурна́я репута́ция. **3.** (*spoilt*) испо́рченный; **go ~** по́ртиться, ис-; **a ~ egg** (*lit.*) ту́хлое яйцо́; (*fig.*) непутёвый челове́к. **4.** (*severe*) си́льный; **I caught a ~ cold** я си́льно простуди́лся; **a ~ wound** тяжёлая ра́на. **5.** (*harmful*) вре́дный; **coffee is ~ for him** ко́фе ему́ вре́ден; **smoking is ~ for one** куре́ние вре́дно для здоро́вья. **6.** (*of health*) больно́й; **I feel ~** я чу́вствую себя́ пло́хо; **be taken ~** (*coll.*) заболе́ть (*pf.*). **7.** (*counterfeit*) фальши́вый. **8.** (*var.*): **a ~ mistake** гру́бая оши́бка; **a ~ debt** безнадёжный долг; **a ~ lot, hat** (*coll.*) дрянь-челове́к; **~ language** ру́гань; **he was in ~ with us** (*coll.*) он был у нас на плохо́м счету́.
cpds. **~-mannered** *adj.* невоспи́танный; **~-tempered** *adj.* раздража́тельный.

Figure 19: Russian (025). Numbering starts over within the entry.

[1]champ [tʃæmp], *v.t., v.i.* geräuschvoll kauen; (*fig.*) mit den Zähnen knirschen; – ***the bit,*** auf die Stange beißen, am Gebiß kauen; (*coll.*) – ***at the bit,*** ungeduldig sein *or* werden.
[2]champ, *s.* (*sl.*) *see* **champion.**

Figure 20: German (003). Integers used to distinguish homographs.

Not all numbers within an entry are necessarily subentry numbers. For example, some dictionaries use "1" to indicate the morphosyntactic attribute "first person".

To help separate the subentry numbers from numbers of other types, a helpful heuristic is to look for consecutive sequences of numbers. If the sequence 1 2 3 1 4 5 is observed, it is unlikely that the second "1" is a subentry number.

Some dictionaries allow the subentry numbering to start over within an entry (this was observed in three dictionaries). For example, the entry in figure 19 contains the numbering sequence 1 2 1 2 3 4 5 6 7 8.

The fact that numbering can start over within an entry must be taken into consideration when designing a heuristic which attempts to automatically identify numbering elements by looking for sequences of consecutive numbers. A sequence such as 1 2 5 6 would presumably always be invalid because of the numbering gap; but a sequence such as 1 2 3 1 2 is potentially valid.

2.1.7 Elements before the headword

20% of the surveyed dictionaries allow some kind of element before the headword. From a processing standpoint, it is fortunate that the elements in question are easily identified. In all of the observed cases, the pre-headword symbols are short, typically consisting of a single letter or a symbol such as an asterisk or dagger. Also, in each dictionary which uses pre-headword symbols, the symbols belong to a small closed set.

Commonly, the element in question is an integer used to distinguish homograph headwords (figures 20 and 21). In

[1]fumi (–) fumi (kind of fish).

[2]fumi (–) *see* **vumi.**

fumo (ma-) 1. *rare* spear. 2. fumo (title of a chief). *Cf.* **[2]fuma, [2]fumbo.**

fumua *inv. of* **[1]fuma.**

Figure 21: Swahili (015). Integers incorporated into cross-references.

امليس *amlas*, Even, smooth, sleek.
آملة *āmula* (S. *āmalaka*), املة *amla*, Name of an excellent eye-medicine; the myrobalan-tree.
A امليس *imlīs*, A barren desert.
A امم *umam* (pl. of *ummat*), People, nations, tribes; sects.

Figure 22: Farsi (005). The 'A' before the headword indicates an Arabic loanword.

the Swahili entry in figure 21, note that the cross-references within the third and fourth entries include the preceding integer. A typical choice during the tokenization process would be to treat the integer and the headword as separate tokens; but both tokens need to be considered when resolving the cross-references during spell-out. This is a complication to the resolution of cross-references.

Some dictionaries use pre-headword elements to indicate something about the etymology of the word. In the Farsi example in figure 22, the **A** before the headword indicates that the word is a loanword from Arabic. Similarly, in an entry from a Latin dictionary (figure 23) , the dagger symbol † indicates that the word is a loanword from Greek.

2.1.8 Headwords

The notion of "headword" is intuitively simple: the headword is the first word in the entry, and it is the citation form for the entire lemma. Unfortunately, this intuition is highly deceptive. Multiple authors (e.g. Neff and Boguraev [20] p. 98; Schneiker et. al. [25] p. 4; Schneiker et. al. [24] p. 84) have remarked on the great complexity of the structure of headwords.

The semantics of the variations on headwords are usually fairly obvious to a human. For the purposes of automated processing, however, the great variety in the form of head-

† **zēlus,** i, *m.*, = ζῆλος, *zeal, emulation; jealousy*, Vitr. 7 praef.; Prud. Ham. 188; Aus. Epigr. 77; Hier. in Gal. 2, 4, vv. 17, 18; Vulg. Num. 25, 11.

Figure 23: Latin (020). The dagger symbol † indicates a Greek loanword.

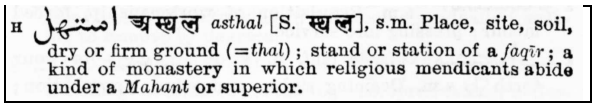


Figure 24: Urdū-Hindi (008). Which elements are part of the 'headword'?

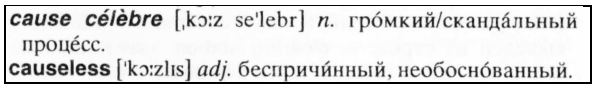


Figure 25: Russian (025). Text attributes in headwords.

words is a significant challenge.

2.1.9 Representation of headwords

Among the dictionaries which make some use of the bold text attribute, nearly all set the headword in bold (85% of surveyed dictionaries make some use of boldface; 78% set the headword in boldface).

The only category of exception is a class of dictionaries where the use of multiple scripts leads to a situation where there is less clarity to the notion of "headword". In the Urdū-Hindi entry in figure 24, which tokens should be tagged as belonging to the headword?

Figure 25 contains a different sort of complication regarding text attributes in headwords. In this Russian dictionary, the headwords are normally in bold but not italic, as in the case of **causeless**. However, in the case of **cause célèbre**, the headword is in both bold and italic, indicating that the phrase can occur in English text but is considered a foreign phrase. Only a tiny minority of entries use italics in this manner, but a program which does not take this attribute into consideration could run into problems.

2.1.10 Disambiguation of homographs

31% of the surveyed dictionaries use an integer to distinguish homograph headwords (**bear**^{1}, **bear**^{2}). Some dictionaries place the integer before the headword; some place it after. Two of the surveyed dictionaries use Roman numerals to distinguish homographs, and one dictionary uses letters.

2.1.11 Dividers within headwords

24% of the surveyed dictionaries permit the headword to contain some kind of divider symbol. The most commonly used divider is the pipe symbol (figure 26; this use is found in 18 out of the 24 dictionaries). In 4 dictionaries, the divider is a slash (figure 27). There are also two dictionaries where the headword divider is a double pipe.

25% of the surveyed dictionaries permit the use of parentheses within the headword to indicate optionality (figure 28). The process of spell-out would need to convert **zombi(e)** to a representation which indicates that both **zombi** and **zombie** are acceptable as spellings of the same word.

Further, in any sort of lexical database, there is usually

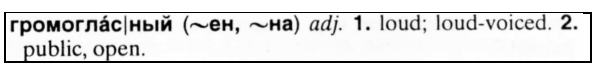


Figure 26: Russian (025). Pipe symbol as a delimiter in the headword.

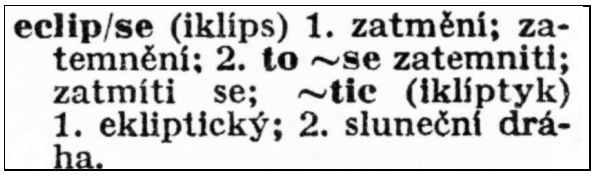


Figure 27: Czech (001). Slash symbol as a delimiter in the headword.

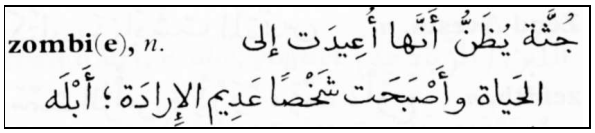


Figure 28: Arabic (007). Parentheses within a headword.

some kind of unique key which stands for an entire lemma. As with the case of forked headwords, to be discussed below, a program which assigns a unique key to each entry or lemma needs to have some way of resolving cases where there is more than one potential label (Neff et. al. [21] p. 86 resolve such cases by creating "multiple entries with cross references"). In cases like **zombi(e)**, the process of spelling out the two alternative spellings may need to happen before the creation of the unique keys, because it may be desirable from a processing standpoint not to permit parentheses within the keys.

2.1.12 Other extra elements in headwords

Beyond dividers such as the pipe symbol, 15% of the dictionaries permit other attributes in the headword which are not a part of the standard orthography of the headword language.

For example, the Italian entry in figure 29 shows the use of dots within the headword to indicate the conventional points where the word may be hyphenated.

Some dictionaries include accent marks to indicate the stress pattern (figure 30).

The Latin dictionary in figure 31 includes marks in the headword to indicate short and long vowels. The headword appears as "advŏcātor", but the same word appears within the entry as "advocator" without the diacritics. This general pattern is commonly observed across dictionaries: there are often mismatches in the writing conventions between the headword and the words in the body of the entry. For example, one dictionary was observed where the headwords are written in all capital letters, but where lower-case letters are used for those same words elsewhere.

This is a complication to certain processing strategies. For example, it would be reasonable to attempt to populate a morphological paradigm by searching for words within the body of the entry which have a low Levenshtein distance

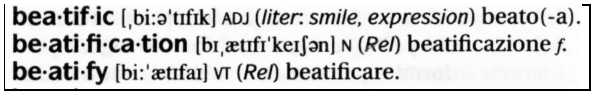


Figure 29: Italian (027). Dots within the headword indicate conventional hyphentation points.

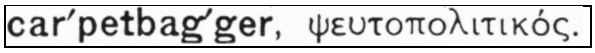

car′petbag′ger, ψευτοπολιτικός.

Figure 30: Greek (002). Accent marks in the headword.

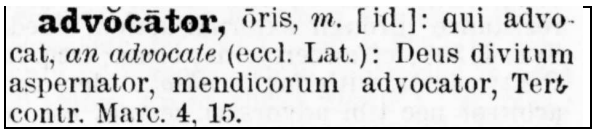

advŏcātor, ōris, *m.* [id.]: qui advocat, *an advocate* (eccl. Lat.): Deus divitum aspernator, mendicorum advocator, Tert. contr. Marc. 4, 15.

Figure 31: Latin (020). Diacritics found in the headword but not in the body of the entry.

from the headword. In the case of the Latin entry, one would probably want to take the additional step of stripping the diacritics from the headword before computing the Levenshtein distances.

2.1.13 *Multi-word headwords*

A headword can consist of a multi-word phrase (figure 32).

Note that the word order in the headword of figure 32 is the conventional word order which one might find in natural text. Often, however, multi-word headwords present the words in an order other than what one would expect in naturally occurring speech or writing (figure 33).

Some dictionaries permit the headword field to contain a lengthy multi-word translation of a single word in the definition language (figure 34). Many of the entries in the Chickasaw dictionary contain lengthy multi-word headwords of the sort cited here. This rather extreme example shows how far the "headword" category can diverge from the intuitive notion of a single-token citation form which is taken to represent all of the inflected members of a paradigm.

2.1.14 *Forked headwords*

Many dictionaries allow the headword field to contain multiple variants, as in figures 35 and 36. In both of these examples, both variants are set in bold. Contrast this with figure 37, where the second variant is not set in bold. A third way of indicating alternate spellings is mentioned by Schneiker et. al. [24], who discuss a dictionary where variant spellings are set in a smaller typeface size than the initial headword.

These forked headwords introduce a couple of complications. First, it is convenient from a processing standpoint to have a single label which stands for the abstract paradigm, or as a reference handle to the entire entry. If there are multiple possible spellings in the headword, then at a minimum, one must arbitrarily pick one of the variants (perhaps the leftmost).

Further, the use of boldface in the Czech and Turkish examples makes it tempting to include the alternate spelling as part of the headword field; but by switching from boldface to non-bold, the author of the Old High German dictionary appears to be communicating that the alternative spelling

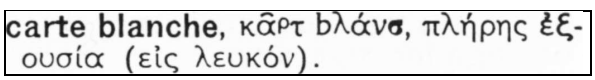

carte blanche, κᾶρτ βλάνσ, πλήρης ἐξουσία (εἰς λευκόν).

Figure 32: Greek (002). A multi-word headword.

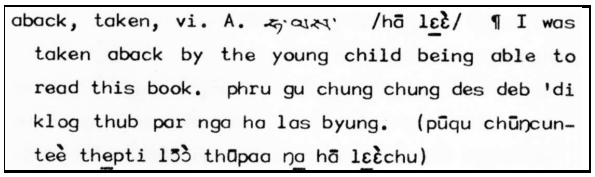

aback, taken, vi. A. རྒྱབ་ལས་ /hā lɛ̱ɛ̀/ ¶ I was taken aback by the young child being able to read this book. phru gu chung chung des deb 'di klog thub par nga ha las byung. (pūqu chūŋcunteè thepti lɔ̄ɔ̀ thūpaa ŋa hā lɛɛ̀chu)

Figure 33: Tibetan (024). A multi-word headword with inverted word order.

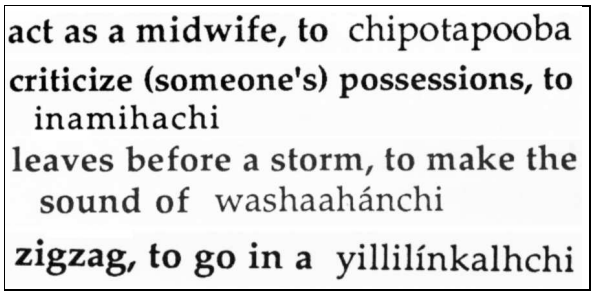

act as a midwife, to chipotapooba
criticize (someone's) possessions, to inamihachi
leaves before a storm, to make the sound of washaahánchi
zigzag, to go in a yillilínkalhchi

Figure 34: Chickasaw (028). Unusually long multi-word headwords.

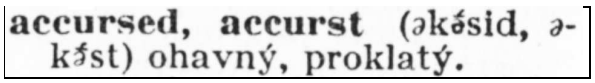

accursed, accurst (ə́kə́sid, ə-kə́st) ohavný, proklatý.

Figure 35: Czech (001). A forked headword.

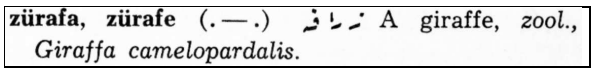

zürafa, zürafe (.—.) زرافه A giraffe, *zool.*, *Giraffa camelopardalis.*

Figure 36: Turkish (035). A forked headword.

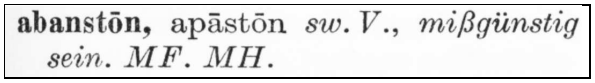

abanstōn, apāstōn *sw. V., mißgünstig sein. MF. MH.*

Figure 37: Old High German (029). Contrast the text attributes with those in the Czech and Turkish examples.

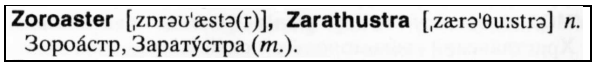


Figure 38: Russian (025). A more complex type of forked headword.

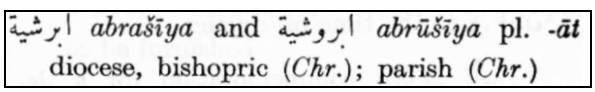


Figure 39: Arabic (031). Another complex forked headword.

should not be considered part of the headword, but rather as part of an "alternative spelling" field. This creates a problem for classification. Should the Old High German entry be tagged in the same way as the Czech and Turkish entries? Do we follow the typographical distinctions and groupings of the authors, or do we impose a preferred conceptual structure? Neither alternative is entirely satisfactory.

A more difficult matter involves forked headwords where the variants can be individually followed by other types of elements (figures 38 and 39). This creates complications on multiple levels. In terms of tagging tokens in the entry by type, should **Zoroaster** and **Zarathustra** both be tagged as type "headword"? If both are tagged as "headword", then for better or worse, this implies a conception of the "headword" field as something which need not be linearly contiguous; or, alternatively, that an entry is not limited to having just one headword.

Neff and Boguraev [20] (p. 98) discuss the difficulty of resolving the scope of the elements which follow the headword variants. To take the Russian entry in figure 38 as an example, each pronunciation field has scope over only one headword variant; but the morphosyntactic abbreviation *n.* (noun) appears to have scope over both branches of the forked headword. Similarly, each transliteration in the Arabic example (figure 39) has scope over only one headword variant, but the notation "pl. *-āt*" evidently has scope over both branches.

A further wrinkle to the issue of forked headwords is that some dictionaries represent the forked structure of the headword with the graphical device of vertically stacking the variant forms (figures 40 and 41).

2.1.15 Transliteration of headwords

16% of the dictionaries include a Roman transliteration of a non-Roman headword (figure 42).

At least one dictionary places the transliteration before the native script (figure 43). There is a fairly clear intuition that the native script is the basic form, and that the Roman transliteration is a derived form which is a convenience to the reader. The headword, which stands for the entire paradigm, normally comes first in the entry; but the Bengali dictionary inverts the expected order.

In five out of the 16 cases, the transliteration is in italics. One case was observed where the transliteration is in all capitals.

2.1.16 Subheadwords

36% of the surveyed dictionaries permit entries to contain one or more subheadwords, as in the Indonesian entry in figure 44.

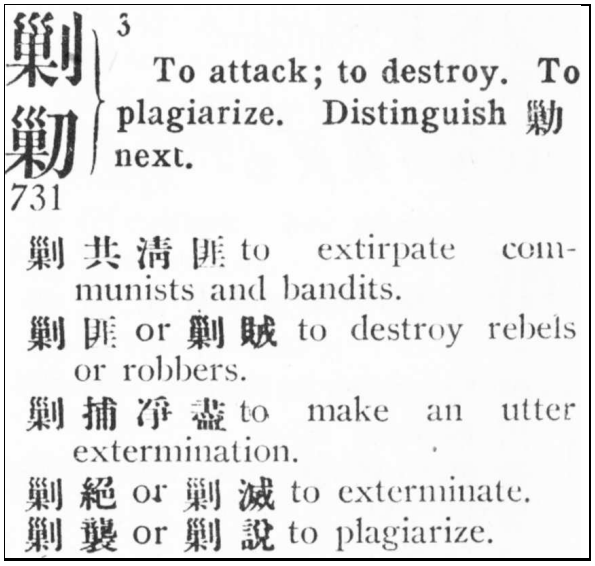


Figure 40: Mandarin (034). Vertical stacking of a forked headword.

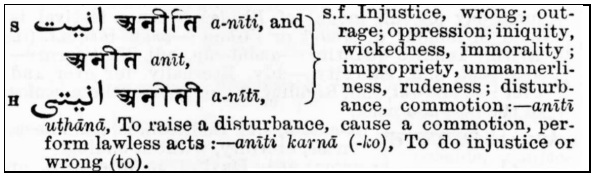


Figure 41: Urdū-Hindi (008). Vertical stacking of a forked headword.

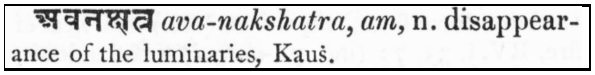


Figure 42: Sanskrit (006). Transliteration of the headword.

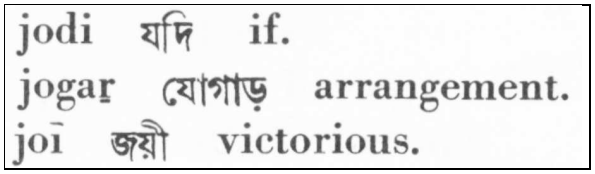


Figure 43: Bengali (011). Transliteration precedes the native orthography.

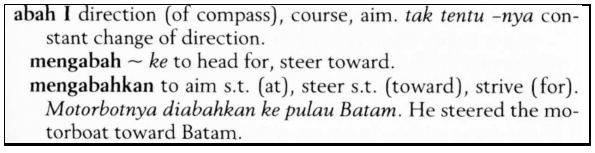


Figure 44: Indonesian (009). An entry containing subheadwords

bacon [ˈbeɪkən] *n.* бекóн; ~ **and eggs** яи́чница с бекóном; (*fig.*): **save one's** ~ спа|сáть, -стú свою́ шку́ру.

Figure 45: Russian (025). Headword ellipsis indicated with a tilde ~

caution [ˈkɔːʃ(ə)n] *n.* **1.** (*prudence*) осторóжность; **with** ~ осторóжно, с осторóжностью. **2.** (*warning*) предостережéние, предосторóжность; **C~!** (*as notice*) Внимáние!; Осторóжно!; **he was let off with a** ~ егó отпустúли с предостережéнием. **3.:** ~ **money** залóг.
v.t. предостер|егáть, -éчь.

Figure 46: Russian (025). The notation 'C~!' indicates that the headword should be capitalized when the ellipsis is resolved.

The subheadword introduces a subentry, whose structure often resembles that of the main entry. In the Indonesian example, both the main entry and the subentry can contain multiple glosses, a usage example, and a translation for the usage example.

30 of the 36 dictionaries mark a subheadword with boldface type.

2.1.17 Headword ellipsis

"Headword ellipsis" is the term used in this article for the substitution of a short symbol for the headword, or for some substring of the headword. Headword ellipsis allows information to be represented compactly by factoring out repeating sequences of characters. The Russian entry in figure 45 uses the tilde ~ symbol as the headword ellipsis symbol.

65% of the surveyed dictionaries include some type of headword ellipsis.

Following is a count of the observed headword ellipsis symbols:

Count	Type	
41	Tilde	~
12	em-dash	—
4	Hyphen	-
1	Two hyphens	--
1	Underscore	_

Some dictionaries have a method of indicating that the elided word should be capitalized (figure 46). Note that the headword **caution** is not capitalized. The notation **C~!** indicates that the headword ellipsis should be resolved as **Caution!** rather than **caution!**.

Three dictionaries use a dot or circle above a tilde to indicate that the word in question should be capitalized (figure 47).

At least one dictionary uses more than one ellipsis symbol. In the Russian entries in (figure 48), a hyphen is used when a suffix is being added to the word, but an em-dash is used when the whole word is standing on its own.

A less common headword ellipsis strategy is to use the first letter of the headword, followed by a period, to stand for the entire headword (figure 49). At least seven of the dictionaries in the sample use this strategy. In one of those dictionaries, the initial letter is followed by a hyphen rather than by a period.

common [ˈkɔmən] zajednički, uobičajen, opći; ~ **Council** općinsko vijeće; ~ **law** običajno pravo; ~ **sense** zdravi razum; ~**er** neplemić; član Donjeg doma; ~ **place** otrcana fraza; banalan; ~**s** *pl* pučani; **House of** ~**s** Donji dom; **the British** ~**-wealth** Britanska zajednica naroda

Figure 47: Serbo-Croatian (059). The circle above the tilde indicates that the headword should be capitalized when the ellipsis is resolved.

Авра́л, *m. naut.* (*a word of command*) all hands on deck!; **-льный,** *adj.* **-льная работа,** work performed by all the crew.
Авра́н, *m. bot.* centaury; **дикий**—, hedge hyssop.

Figure 48: Russian (017). The dictionary uses two symbols for headword ellipsis.

2.1.18 Morphological fragments

20% of the surveyed dictionaries represent information about inflected forms by means of hyphenated fragments (figures 50 and 51).

For example, consider the Italian entry in figure 50. Note the use of **-er, -est,** which encodes that the comparative and superlative forms are **coarser** and **coarsest,** respectively.

Three of the surveyed dictionaries use a tilde instead of a hyphen to indicate morphological fragments, as in the Hopi example in figure 52.

One dictionary uses a plus sign + to indicate morphological fragments. Two dictionaries include morphological fragments without any hyphen or other punctuation. This means that a token tagging program must identify these tokens without the help of the informative "initial hyphen" attribute. A reasonable heuristic is to make a list of known, frequently recurring fragments, and to annotate tokens as belong to that list.

2.1.19 Resolution of headword ellipsis

Resolution of headword ellipsis is a very complicated matter. Consider the Russian entry in figure 53. The ellipses in this entry would presumably be resolved as Австри́йка and Австрали́йка. This resolution involves a substitution: first something is removed from the headword, and then something is added. This notation is potentially ambiguous: for example, if there were two л characters at different positions within the headword, how would -ли́йка be resolved? (If a Russian corpus is available, a possible approach is to compute both possible resolutions, and to see whether only one

виргúнский *adj.*: **в. табáк** Virginia tobacco.

Figure 49: Russian (025). Headword ellipsis is indicated with the initial letter followed by a period.

coarse [kɔ:s] ADJ (*comp* **-er**, *superl* **-est**) (*texture, skin, material*) ruvido(-a); (*salt, sand*) grosso(-a); (*sandpaper*) a grana grossa; (*vulgar: character, laugh, remark*) volgare.

Figure 50: Italian (027). -er and -est are morphological fragments.

kù-**zuukira** *v.i.* (**-dde**) rise from dead; be restored to life. *kù*-**zuukiza** *v.tr.c.* (**-izza**) raise from dead; restore to life.

Figure 51: Luganda (036). -dde and -izza are morphological fragments.

a'kima (~**ya**) *vt.p.pl.obj.* have been to pick corn. *Nu' pöma'uyit sòosok pu'* ~. I've been to pick all the corn off the stalks of the early crop. *a-'ki-ma* [RDP-pick:corn-POSTG]

hopi|**himu** (~**hìimu**; *acc.* ~**hìita**) *n.* Hopi things. *Nu' as ung hikikw hìita* ~***hìita*** *tuuvingtani.* I'd like to ask you about some Hopi things. — ~***hìimu*** *naat a'ni poninita.* Hopi things are still viable. *hopi-himu* [Hopi-thing]

Figure 52: Hopi (021). Use of the tilde ∼ to indicate morphological fragments.

Австри́ец, *m.* **-ри́йка,** *f.* an Austrian.
Австрали́ец, *m.* **-ли́йка,** *f.* an Australian.

Figure 53: Russian (017). Difficulties in the resolution of headword ellipsis.

Ast [ast], *m.* (**-es**, *pl.* ¨**e**) 1. bough, branch, limb (*of tree*); 2. knot (*in wood*).

Figure 54: German. Resolution of headword ellipsis requires language-specific knowledge of Umlaut.

вин|**о́, á,** *pl.* ~**a** *nt.* **1.** wine. **2.** (*sg. only*; *coll.*) vodka.

Figure 55: Russian (025). Resolution of headword ellipsis requires language-specific knowledge of shifting accent.

is found in the corpus.)

Spell-out can require expert knowledge of the morphology of the language. In figure 54, the notation "¨**e**" means here that the plural of **Ast** is **Äste**. The resolution of headword ellipsis in this dictionary requires string substitutions which are highly specific to German.

A similar case is found in the Russian example in figure 55. The acute accent over the tilde indicates that the stress shifts in the inflected form, i.e. the inflected form is **ви́на**. Once again, language-specific expertise would be required to code a program which can resolve this type of headword ellipsis.

Figure 56 contains a different kind of complication to the resolution of headword ellipsis. Note that there are two instances of ellipsis in this entry:

— ***garden*** should be resolved as ***zoological garden***

[–ˈɔlədʒist] should be resolved as [zouˈɔlədʒist]

Thus, there are two different repeating strings which have been factored out here: one is the headword, and the other is a portion of the pronunciation. A program to resolve headword ellipsis cannot simply substitute the headword in every case where the ellipsis symbol is found; the field type must be taken into consideration.

Because of the highly language-specific nature of headword ellipsis resolution, a full general solution is probably not possible. This is an area where each individual dictionary will probably always require human intervention.

2.1.20 Morphosyntactic markers

77% of the surveyed dictionaries include some kind of abbreviations indicating morphosyntactic information such as part of speech (noun, verb), subcategory (intransitive verb) or morphosyntactic features on an inflected form (past participle).

Out of these 77 dictionaries, the morphosyntactic abbreviations are in italics in 58 dictionaries. The abbreviations are followed by a period in 40 of the dictionaries.

There are also other, less common ways of denoting morphosyntactic abbreviations. 6 dictionaries have the abbreviations in parentheses. 3 have the abbreviations in all up-

zoological [zouəˈlɔdʒikl], *adj.* zoologisch; – ***garden,*** zoologischer Garten, der Tiergarten. **zoologist** [–ˈɔlədʒist], *s.* der Zoologe (die Zoologin).

Figure 56: German (003). Ellipsis in both headword and pronunciation.

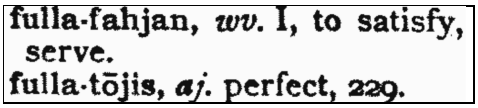

fulla·fahjan, *wv.* I, to satisfy, serve.
fulla·tōjis, *aj.* perfect, 229.

Figure 57: Gothic (073). The contents of the morphological class fields are simple and occur with high frequency.

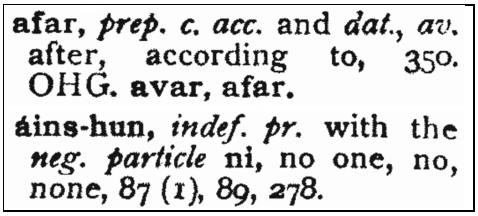

afar, *prep. c. acc.* and *dat.*, *av.* after, according to, 350. OHG. avar, afar.
áins-hun, *indef. pr.* with the *neg. particle* ni, no one, no, none, 87 (1), 89, 278.

Figure 58: Gothic (073). Arbitrary prose within the morphological class fields.

percase, and out of those, two do not have a following period. One dictionary has the morphosyntactic abbreviations in boldface.

2.1.21 Complex structure within ranges of morphosyntactic markers

Within a single dictionary, the level of complexity of a range of morphosyntactic information can vary. Consider the two entries in figure 57.

"*wv.* I," and "*aj.*" are typical of the ranges of morphosyntactic information in this dictionary; a relatively short list of strings such as these two would give coverage of the morphosyntactic ranges for the great majority of entries in the dictionary. However, a small minority of these ranges have a greater complexity (figure 58). The following two ranges can be considered morphosyntactic information:

> *prep. c. acc.* and *dat., av.*
>
> *indef. pr.* with the *neg. particle* **ni,**

These fields contain a mix of standardized abbreviations interspersed with fragments of English prose. A nearly arbitrary level of complexity appears to be permitted.

This is an instance of a more general problem. Typically, a fairly modest amount of effort provides coverage for a majority of cases, but additional effort gives only a small gain. In the case of the Gothic dictionary, one could readily construct a list of complete morphosyntactic ranges such as "*wv.* I," which would cover the great majority of entries; but there is a residue of cases such as the two complex cases cited here which do not lend themselves to any simple, general kind of machine-readable representation.

2.1.22 Frequently used abbreviations

It was informally observed that certain abbreviations such as *adj., n.,* and *vi.* are found in many dictionaries. These abbreviations are not universal across all dictionaries, but they are very common. This is true not only of morphosyntactic abbreviations, but also for domain indicators such as *med., chem., mus.* Although it was not undertaken for the current work, a potentially useful project would be to compile a collection of tables of abbreviations over a large number of

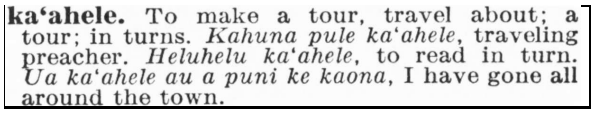

ka'ahele. To make a tour, travel about; a tour; in turns. *Kahuna pule ka'ahele*, traveling preacher. *Heluhelu ka'ahele*, to read in turn. *Ua ka'ahele au a puni ke kaona*, I have gone all around the town.

Figure 59: Hawaiian (012). Usage examples.

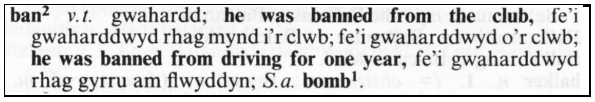

ban[2] *v.t.* gwahardd; **he was banned from the club,** fe'i gwaharddwyd rhag mynd i'r clwb; fe'i gwaharddwyd o'r clwb; **he was banned from driving for one year,** fe'i gwaharddwyd rhag gyrru am flwyddyn; *S.a.* **bomb**[1].

Figure 60: Welsh (018). Usage examples.

dictionaries. This information can potentially be leveraged:

- Some dictionaries are published without a table of abbreviations, and the meaning of all of the abbreviations is not always apparent. Comparison with other dictionaries can potentially help resolve some of the unclear abbreviations.
- In terms of doing an initial survey on a dictionary, a program can search for commonly used abbreviations, both to determine whether such abbreviations are present, and also to determine what text attributes are used. If *adj.* is italic and followed by a period, then other words with those attributes are potentially also morphosyntactic abbreviations.

2.1.23 Pronunciation

43% of the dictionaries contain a pronunciation field. Out of these 43, a majority (36 dictionaries) enclose the pronunciation in some kind of delimiter:

Count	Type
26	Brackets
4	Parentheses
4	Slashes
2	Other delimiters

23 of the 43 dictionaries indicate the pronunciation in the International Phonetic Alphabet, or in a form of the IPA with some non-standard variation such as the use of acute accents to indicate stress. Obviously, it would be useful to build a general-purpose tool which attempts to identify pronunications on the basis of the presence of IPA-specific characters such as ʃ or ɔ.

There was one unusual case observed where a dictionary contains a field indicating only the stress pattern but not the segmental pronunciation.

2.1.24 Gloss

The gloss in the translation language is most often in plain text. In 7% of the dictionaries, it is in italics.

2.1.25 Usage examples

70% of the surveyed dictionaries contain usage examples (i.e., phrases or sentences containing uses of the headword, usually with accompanying translations; figures 59 and 60).

26 of these 70 dictionaries use bold for the headword language within the usage example, as in the case of the Welsh dictionary. Five dictionaries have the translation language in italic. There are also dictionaries which use both bold

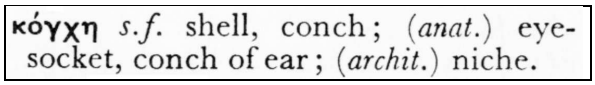


Figure 61: Greek (039). Domain indicators.

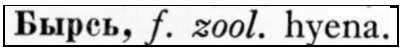


Figure 62: Bulgarian (017). Domain indicator.

and italic together for the portion of a usage example which is written in the headword language.

2.1.26 Domain indicators

41% of the surveyed dictionaries contain domain indicators such as *mus., med., chem.,* etc. (figures 61, 62, and 63). In the case of figure 63, note that the Chinese character 空 (lit. "sky") is a domain indicator referring to aviation.

Out of these 41 dictionaries, 34 set the domain indicator in italics. 19 enclose the domain indicator in parentheses, and 18 have a period following the domain indicator abbreviation.

2.1.27 Sense indicators

69% of the surveyed dictionaries include sense indicators. The Navajo entry in figure 64 contains sense indicators: "(for weaving)" and "(story, tale)" distinguish different senses of "yarn".

61 of the 69 dictionaries enclose the sense indicators in parentheses. 35 of the 69 dictionaries set the sense indicators in italics.

There are two dictionaries where it was observed that some sense indicators are enclosed in square brackets, but where other sense indicators are enclosed in double parentheses. The meaning of the distinction, if any, was not clear.

2.1.28 Etymology

8% of the surveyed dictionaries contain an etymology field. Of those 8, the etymology is enclosed in square brackets in 4 of the dictionaries.

As noted in the case study of the Old English dictionary, the identification of etymology fields poses special problems. An etymology field often contains tokens of diverse types, some of which may be confusable with other token categories.

Because of this problem, it was observed in the Old English case study that the CRF often failed to correctly identify the beginning and end of the etymology field. This problem was remedied by writing a custom script to identify the etymology field by searching for diagnostic abbreviations within ranges enclosed by square brackets; each token within the square bracket range was tagged with the value 1 for an "etymology" attribute.

For most NLP applications, the etymology field is unlikely to be of much practical use. The approach taken in the Old English case study is to treat the etymology field as a single

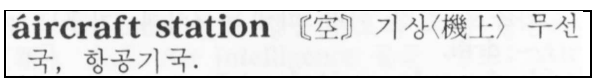


Figure 63: Korean (045). Domain indicator.

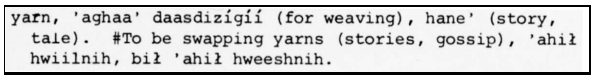


Figure 64: Navajo (043). 'for weaving' and 'stories, gossip' are sense indicators.

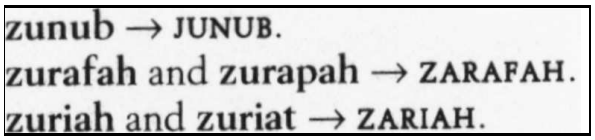


Figure 65: Indonesian (009). Cross-references.

field, deliberately overlooking the diversity of token types which are typically found in an etymology.

2.1.29 Cross-references

28% of the dictionaries permit the entry to contain a cross-reference to another entry (figures 65 and 66)

6 out of the 28 dictionaries set the cross-referenced word in bold.

It was informally observed that a variety of symbols or abbreviations is used across dictionaries to indicate cross-references. Following is an indication of some of the major patterns:

- At least 8 dictionaries use an equals sign = to introduce a cross-reference.
- At least 6 dictionaries introduce a cross-reference with some variant of "see" ("See", "see", or "s.").

3. DISCUSSION

3.1 On the problem of "irregular" entries

From a processing standpoint, the easiest sort of dictionary would be one where all entries conform cleanly to some concise and clearly defined entry grammar. In the real world, there are few dictionaries, if any, which meet this ideal. Nearly every author who has written on the subject of processing dictionary text has remarked on the problem of irregular entries [21] [6] [13].

In discussing irregular entries, most authors have assumed a view where there is a clear two-way division between regular and irregular entries. For example, the TEI guidelines [26] include tags for the markup of dictionary text. The tag set includes <entry> for entries which conform to some well-defined grammar of entries. To accommodate irregular entries, TEI also includes an <entry-free> tag, which permits entry elements to be freely combined in any order. The TEI standard thus formally enshrines a conception where all dictionary entries fall fully into either a "regular" or "irregular" category.

The literature often describes irregular entries using words which carry connotations of badness. For example, Lem-

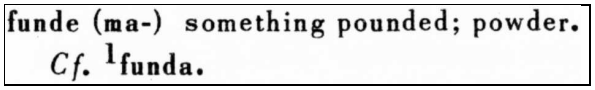


Figure 66: Swahili (015). Cross-references.

decade, neo. བརྒྱ་ཚོ་ /cūlo/ ¶ two decades · bcu lo gnyis (cūlo ཉིས) [Note: Normally said simply as "x" no. of years, e.g., 2 decades is 20 years.]

Figure 67: Tibetan (024). The 'Note' element is found only in this entry.

"zamindar"/jomidar জমিদার *n.* land owner.

Figure 68: Bengali (011). The headword is of an idiosyncratic form not found elsewhere in the dictionary.

nitzer and Kunze [17] describe irregular entries as "malformed".

Many authors speak of dictionaries as having a grammar for the form of the entries [10] [17]. This could be taken to mean that the original author of the dictionary is assumed to have had a grammar in mind when writing the dictionary; or it could merely mean that an after-the-fact construction of a grammar should be considered a useful step in processing dictionary text. Further, the notion of dictionary entries having a hierarchical structure is widely assumed in the literature (see, e.g., Neff et al. [21] p. 84; Klavans and Tzoukermann [15]; Kammerer [13] p. 10).

Ide et. al. [11] develop this notion fully. The specific concern of Ide et. al. is not the parsing of existing dictionaries; rather, it is the creation of new dictionaries. In the schema of Ide et. al., a dictionary has an explicitly defined context-free grammar specifying the form of entries. Further, the model includes a well-defined way of handling attributes on nodes in the hierarchical parse tree. A child node implicitly inherits all of the attributes of its parent. However, a child node is permitted to explicitly overwrite the parent's value for an attribute. If a child node overwrites a value, then all of the descendants of that node inherit the new value.

Few (if any) human-readable dictionaries live up to this ideal of full regularity. With this observation in mind, it is worth examining the notion of regularity more critically.

Dictionaries do not come with explicitly published grammars. This means that a grammar for a dictionary must be inferred by observing the patterns in the dictionary entries (cf. Hauser and Storrer [10] p. 7, who discuss how a tool set can be designed to help the investigator infer a grammar from the dictionary entries). From a cursory examination of a dictionary, it is usually a fairly simple matter to write out a grammar which matches the most frequently occurring entry types. When the entire dictionary is parsed against this grammar, however, there are always entries which fail to match the grammar.

Consider the entries in figures 67 and 68. The Tibetan entry is the only one in the study sample which includes the "Note:" constituent. Similarly, the headword in the Bengali example is the only one in the study sample which has this highly specific form (two variants separated by a slash, with one of the variants enclosed in curled quotes).

The grammars for these dictionaries could be made to accommodate these elements. This might have the dubious consequence that a subrule is added to the grammar to handle a single entry in the entire dictionary. Adding a single

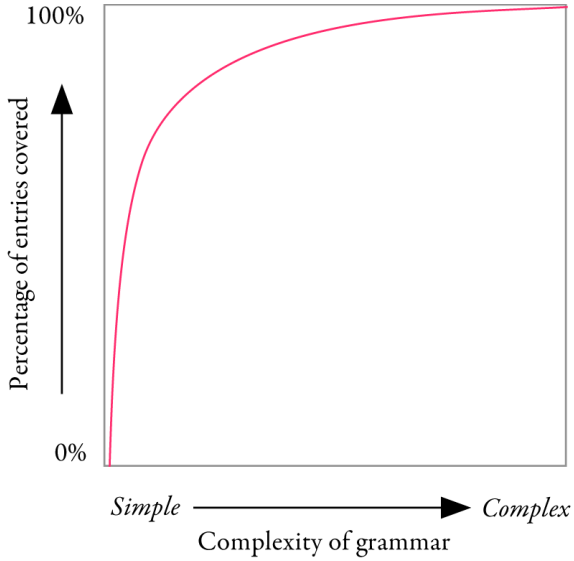


Figure 69: Relationship between time/effort and completeness of coverage of an entry grammar

subrule of this kind does not involve much effort, but it is impractical to handle hundreds of low-frequency or unique elements in this way. Further, even if the practical limitations on time and effort are ignored, the addition of low-frequency elements to a nonprobabilistic grammar increases the probability that entries of high-frequency types could be incorrectly parsed (Kammerer [13] p. 22-3).

Since a dictionary is finite in length, it is always possible to formulate a grammar which exhaustively generates every entry in the dictionary. Assuming that the dictionary tokens have already somehow been tagged by type, producing a comprehensive grammar for a dictionary is simply a matter of iterating through all of the entries, and listing out a disjunctive rule which includes every unique observed sequence of token types:

entry → *headword pos pos def def def*
entry → *headword alt-headword pos def def def def*
entry → *headword pos cross-reference cross-reference*
etc.

From this viewpoint, an "irregular entry" merely means "an entry which my grammar does not handle".

There is not usually a great deal of effort involved in writing a grammar which handles the majority of the entries. By contrast, handling the less frequent entry types is a matter of chasing diminishing returns. This relationship between completeness versus time/effort can be visualized as in figure 69.

There is no obvious point where the effort should stop. It is sometimes acknowledged in the literature that the decision as to what should be included in the grammar is an arbitrary one:

> "Entries which are too irregular to fit into such a class should be defined as irregular. *The boundary between regularity/irregularity should be defined* by the database manager (linguist) and hence be unalterable by lexicographers. Irregular entries are

therefore defined in the conceptual schema (the interactivity of the interface, the powerful editing functions and the incremental compilation provide for the feasibility of this approach)." (Domenig and Shann [6] p. 94; emphasis added)

3.2 On the nature of human-readable dictionaries

In the preceding section, I argued that a full regularity is not to be found in human-readable dictionaries. In the present section, I discuss why this should be the case.

Consider the category of nouns. Any human language contains thousands of nouns. Further, many linguists would find it reasonable to suppose that there is a psychological reality to a morphosyntactic feature [±noun]. The nature of this lexical information is such that it lends itself well to being succinctly encoded in dictionary entries, e.g. with the abbreviation "*n.*"

Not all information about lexical items is of this sort, however. Consider the Kikuyu entries in figure 70. These entries contain the following substrings:

> "sometimes used by girls in addressing their mother"

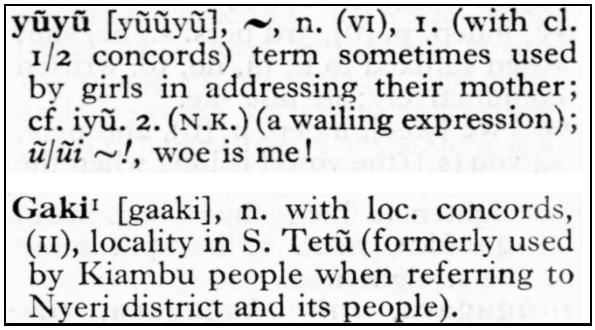
> "formerly used by Kiambu people when referring to Nyeri district and its people"

This is also a kind of information about the lexical items, but it is pragmatic or sociolinguistic information which does not lend itself to analysis in terms of features such as [±noun]. This kind of information is best handled by permitting a dictionary entry to contain ranges of arbitrary prose.

Both kinds of information are of interest to the human user of the dictionary. This means that highly structured information must be intermixed with fundamentally unstructured information.

I claim that the semi-regular nature of human-readable dictionaries follows from two facts:

1. Dictionaries are written for humans. More specifically, the author of a dictionary can reasonably assume that the user of the dictionary has access to the full range of human cognitive abilities when interpreting an entry. ("The problem with dictionary and encyclopedia entries is that, although they are constructed in a principled manner over many years by professional lexicographers and encyclopaedists, they are designed for human use." [29])
2. It is true that much lexical information can be reduced to a fixed form or a closed set of primitives, such as a fixed set of part-of-speech tags. However, there is some lexical information which cannot reasonably be reduced in this manner and is best expressed through prose.


yũyũ [yũũyũ], ~, n. (VI), 1. (with cl. 1/2 concords) term sometimes used by girls in addressing their mother; cf. iyũ. 2. (N.K.) (a wailing expression); *ũ/ũi ~!*, woe is me!

Gaki[1] [gaaki], n. with loc. concords, (II), locality in S. Tetũ (formerly used by Kiambu people when referring to Nyeri district and its people).

Figure 70: Kikuyu (037). Some of the information about the lexical items is expressed as arbitrary prose.

The conception which is adopted here is that the author of a human-readable dictionary is permitted to switch freely between structured information and prose. This is not a defect; "irregular" entries are not "malformed" from the perspective of the human author or user. Rather, human-readable dictionaries are optimized for their intended primary purpose, which is to provide a human user with useful information about lexical items. The dictionary author chooses whatever strategy is most appropriate: if the information lends itself to being expressed in a regular form, then the author chooses the regular form; but the author freely resorts to arbitrary prose when the information requires it.

The distinction between structured and prose information is made more complex by the fact that prose and structured information can be freely intermingled. Recall the discussion from section 2 regarding ranges of morphosyntactic information in a Gothic dictionary. The majority of the morphosyntactic ranges occur multiple times, belonging to a set of entries such as "*wv.* I" (weak verb of the first class) or "*sv.* IV" (strong verb of the fourth class). One could list a fairly short set of such strings of abbreviations which would cover the great majority of these entries. However, it was noted that there is a minority of these morphosyntactic ranges where the information is idiosyncratic and more complex:

> *prep. c. acc.* and *dat., av.*

> *indef. pr.* with the *neg. particle* **ni,**

These strings appear in the same position within the entry where one would normally expect an ordinary set of morphosyntactic symbols, such as "*wv.* I". These ranges can be thought of as free prose which embeds abbreviations such as *acc.* or *neg.*

Dictionaries seem to hover on the verge of regularity, but never quite achieve it. The amount of irregularity which one finds in a dictionary follows from the nature of lexical information; it represents the relative amounts of lexical information which can best be expressed by these two respective means.

This way of thinking about dictionaries has consequences for how the processing of dictionaries should be handled. Creating NLP resources from human-readable dictionaries is a matter of producing something regular from something which is fundamentally irregular. Trying to exhaustively parse a dictionary is futile.

This observation has been made by others. Neff and Boguraev [20] (p. 94-5) note that dictionaries typically contain entries including elements which do not follow any general pattern or set of rules; they describe one dictionary in which usage notes "can be arbitrarily complex and unstructured fragments, combining straight text with a variety of notational devices (e.g. font changes, item hilighting and notes segmentation) in such a way that no principled structure may be imposed on them." They state that parsing should therefore be conducted in a way which can fail "gracefully".

Some authors have expressed skepticism about using human-readable dictionaries to produce resources for NLP (van der Eijk et. al. [27] p. 53-4). I take a different view; dictionaries can be useful as a source of data for NLP applications, but there are limitations which must be kept in mind:

- There is no "truth" in terms of a correct grammar of dictionary entries. The act of processing dictionary entries requires the human investigator to make decisions about what formal categories will be imposed upon the semiregular data.

- There is no clearly defined point where one is "done". There comes a point where extracting further information from the dictionary is more expensive than creating the same information by hand. This point exists, but there is no easy way of knowing that it has been reached.

- Some information must be left behind. The NLP lexicon is required to have a regular form, and there will always be some information in the dictionary which cannot be shoehorned into that form. The task of processing the human-readable dictionary is not so much a matter of converting one file format to another; it is more like smelting metal from a raw ore (albeit a high grade ore, as contrasted with the strategy of inducing lexical information from natural language corpora; see Gopestake et. al. [8] p. 184).

- It is not necessarily useful to try to produce a complete hierarchical grammar of the entries of a dictionary. Since prose can be freely inserted at any point in the entry as needed, a fairly large number of entries will contain at least some irregularity. On the other hand, entries seldom consist entirely of arbitrary prose; most contain at least some structured sections. Processing dictionary entries is therefore not a matter of fully parsing each entry, but rather of identifying subranges of entries which can be parsed.

  "[A] dictionary entry is a typical instance of structured information... However, since the structure of a dictionary entry is relatively flexible too, we need to be careful not to be too strict in modeling the structure." (Kang [14] p. 226)

3.3 Stages of producing a parsed dictionary

In section 1, I considered the problem of tagging tokens in dictionary entries. Token tagging is just one of the steps involved in dictionary processing. In this section, I will attempt to formalize a broader and more complete sequence of stages.

Various stages have been identified in the literature on dictionary processing. The current work generalized over this literature by distinguishing the following six stages:

1. **Acquisition:** production of a clean, normalized digital form of the raw text, including an encoding of presentation-level text attributes such as boldface and italics

2. **Tokenization:** segmenting the stream of text into a sequence of meaningful elements

3. **Token tagging:** assigning a category to each token; this is similar to part-of-speech tagging in the processing of natural language data

4. **Parsing:** recovery of a higher-order structure, typically a hierarchy, allowing the scope of semantic elements and attributes to be explicitly encoded

5. **Spell-out:** a catch-all term for various types of post-processing, typically involving the conversion of the highly compressed dictionary notation into a fully expanded form. For example, given the input "Auto pl. -s", the process of spell-out might produce a representation explicitly encoding that the form *Auto* is singular and that the corresponding plural form is *Autos.*

6. **Validation:** confirmation that the resulting representation of the data conforms to a set of expected norms

Each one of these distinctions has some kind of antecedent in the literature. For example, Neff and Boguraev [20] distinguish the stages which I refer to as tokenization, token tagging, and parsing (but also make statements which explicitly collapse the stages which I distinguish here as parsing and spell-out). Lemnitzer and Kunze [17] distinguish what I refer to as tokenization, parsing, spell-out, and validation, albeit with some differences in terminology (they use the term "post-processing" for what I am calling spell-out, and "consistency checking" for validation). Further examples of this kind could be cited.

The six stages represent a useful classification for talking about the different types of activity. In practice, the division between stages is not so tidy. In particular, there is often feedback from later stages to earlier stages in the form of error correction. For example, tokenization or tagging can reveal errors which were not caught in the acquisition stage:

> "Dictionary entry grammars define conditions for well-formedness of dictionary entries and specify partitive and precedence relations between the constituents of the dictionary entry structure. Dictionary entry structures not licensed by the dictionary entry grammar will be marked as non-wellformed. As a consequence, the process of dictionary entry parsing—aside from its main goal of converting the typesetting tape into a lexical database—has the side-effect of detecting errors and inconsistencies in the structural encoding of the dictionary." (Hauser and Storrer [10] p. 1)

Each of the stages will be individually discussed below.

3.3.1 Acquisition

Acquisition refers to whatever preliminary steps are required to produce an **initial text,** which serves as input to tokenization.

Acquisition has been approached in various ways. Some projects have made use of paid typists [7]. Some have used OCR followed by hand-correction [18]. Some have processed the raw electronic typesetting file acquired from the dictionary publisher [20].

Producing an initial text often involves various kinds of normalization. The character encoding can be standardized (e.g. to UTF-8). The representation of presentation-level

attributes such as bold can be converted to some standard form (e.g. HTML-style tags such as ...).

In the 1980's, a number of dictionaries became available to researchers in the form of typesetting files provided by publishers. In particular, there is a string of articles discussing the *Longman Dictionary of Contemporary English* (LDOCE). Alshawi et. al. [3] discuss some of the details of working with this resource. For example, it was determined that some of the control codes embedded in the text were present purely for aesthetic typesetting reasons and did not encode anything about the semantic structure of the dictionary. Handling these control codes is a part of the acquisition process.

In the case of the Lau dictionary discussed in section 1, the raw text was simply downloaded from Project Gutenberg. Further acquisition activity included normalizing the markup information for text attributes such as italics.

Following is a suggested partial list of checks to be carried out during acquisition:

- Unbalanced markup tags have been resolved.
- Whitespace has been normalized (after significant layout, such as narrower margins, has been encoded by other means).
- In cases where the layout of the entry is a structure indicator, symbols have been embedded in the initial text to preserve this information.
- Character encoding has been normalized (current best practice would typically be to normalize the text to UTF-8). A histogram of characters can be created to detect erroneous or unexpected characters.

3.3.2 Accuracy

Digitization over non-trivial amounts of text is never 100% accurate, even if it is carefully checked by humans. This inaccuracy should be taken into consideration in the design of the downstream processes of tokenization, token tagging, and parsing.

Most digitized dictionaries contain errors; some of these errors might exist in the original text, and others might be introduced during acquisition. An approach which is not designed with errors in mind is likely to be brittle. For example, an OCR program might misrecognize a period as a comma; if a monolithic script assumes that part-of-speech tags are followed by a period and not a comma, the script is less likely to produce the desired output. This is an argument in favor of statistical approaches such as the use of a CRF; an approach which looks at multiple factors in terms of probabilities will have a general tendency to be more resilient to small errors.

For everyday OCR tasks, such as the recognition of newspaper text, accuracy can be greatly improved with a word model or language model (in English, "the" is much more common than "tbe"). Unfortunately, when performing OCR on a dictionary involving a less commonly taught language, it is often the case that no such models are available. It might be the digitization of the dictionary itself which produces the initial word-list which can subsequently be used to help with OCR over texts in the language in question. This means that there may be no alternative to laborious hand-correction by humans, although a corpus might be leveraged.

3.3.3 Line-final hyphens

The issue of line-final hyphens is another challenge in the acquisition stage. Consider the following artificially constructed English paragraph:

> Mike has a healthy self-
> confidence when he talks
> about physics. His under-
> standing of the subject
> matter is obviously good.

The word "self-confidence" is conventially written with a hyphen even if the entire word appears on one line, but the hyphen in "understanding" is an artifact of the line break. Line-final hyphens are ambiguous. In the case of German, there is a three-way ambiguity; in addition to words which do and do not ordinarily contain hyphens, German also permits words to end with hyphens (*ein- und auslaufen*).

In the case of natural language text, there are obvious strategies for repairing line-final hyphens. In any reasonable English corpus, the words "self-confidence" and "understanding" can be expected to be much more common than "selfconfidence" or "under-standing". For text in natural languages such as English, line-final hyphens can be repaired with very good accuracy simply by picking whichever line-internal variant is more commonly found.

Unfortunately, dictionaries of lesser-taught languages do not lend themselves well to this hyphen repair strategy. First, a suitable corpus may not exist. Second, bilingual dictionaries by definition mix data from at least two languages, which complicates any use of corpora. Third, dictionaries often include hyphenated word fragments which should not be joined to anything. For example, many dictionaries contain hyphenated word fragments to show something about the morphology of the word. There appears to be no simple solution to this general problem.

3.3.4 Tokenization

Tokenization is the process of segmenting the stream of characters into substrings, each of which is an individual symbol. In the case of computer languages, tokenization is often accomplished with a finite state machine [1].

Exactly what should be considered a "token" is a question to which there is no single correct answer. Consider the Italian entry in figure 71. Here are two of the ways that this entry might be tokenized:

Strategy 1:	**Strategy 2:**
thith•er	thith•er
['ðɪðər]	[
ADV	'ðɪðər
(old,	]
liter)	ADV
là,	(
laggiù	old
	,
	liter
	)
	là
	,
	laggiù

The first strategy is simply to tokenize on whitespace; this is the approach used in section 1 in the case studies on

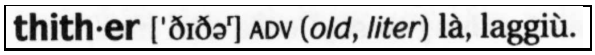


Figure 71: Italian (027). There is more than one way to tokenize the entry.

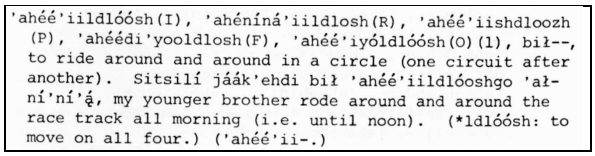


Figure 72: Navajo (043). Apostrophes should not be treated as separate tokens.

CRFs. The second strategy separates punctuation marks from their host words and treats them as separate tokens. This second strategy is essentially the one adopted in the LexParse dictionary parsing system [10].

Tokenizers for natural language often treat punctuation marks as separate tokens [12] (p. 194). There are some potential pitfalls in implementing this approach for dictionary processing.

Consider the use of apostrophes in Navajo (figure 72) and Samoan (figure 73). In many languages, the straight apostrophe or curled single-quote characters are used as punctuation marks, and would generally be treated as separate tokens; but in languages such as Navajo and Samoan, these characters can be part of the orthographic representation of a word. For example, the Samoan word **'a'asa** should be treated as a single token; it would be a mistake to separate the word into multiple tokens due to the presence of the curled single quote.

3.3.5 Token tagging

Token tagging is the association of a category label with each token. Because this problem was the main focus of section 1, it will not be discussed here.

3.3.6 Parsing

Parsing is the process of assigning a hierarchical structure to the tokens of an entry. This step is similar to syntactic parsing of natural language text or the parsing of a computer program.

Consider the Greek entry in figure 74. As discussed, token tagging is concerned with assigning a category to each token. The Greek entry might be tagged as follows:

κοινωνῶ	HEADWORD
1.	SECTION-NUMBER
v.i. & t.	POS
etc.	

By contrast, the process of parsing groups these elements into larger structures. Consider these two substrings within the entry:

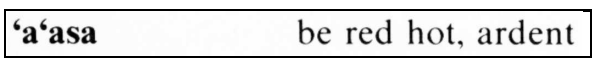


Figure 73: Samoan (013). Apostrophes should not be treated as separate tokens.

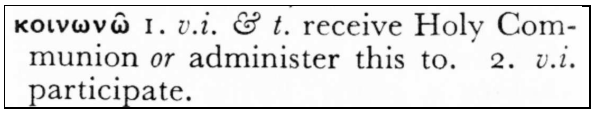


Figure 74: Greek (039). Hierarchical organization of elements.

1. *v.i. & t.* receive Holy Communion *or* administer this to

2. *v.i.* participate

If the structure of the entry is expressed as a hierarchy, then each of the two preceding units might represent the terminal nodes beneath a "subentry" node. Some elements have a restricted scope which can be stated in terms of this kind of hierarchy. A reasonable interpretation of the preceding entry is that each set of morphosyntactic abbreviations (*v.i.*) has scope over its parent subentry node, and also over all of the descendants of that subentry node; but this scope does not extend outside of the subentry.

Above, I discussed at length the problem of inferring a concise grammar from a document which is fundamentally irregular in its structure. An approach which attempts to exhaustively parse every token in every entry, according to some restricted grammar, is likely to fail on a substantial percentage of entries.

Neff et. al. [21] parsed the Collins English-German dictionary using a top-down parser, and achieved a successful parse for only around 80% of the 46,000 entries. For the Collins Italian-English and English-Italian dictionaries, the success rate was higher (around 95%). They attribute this difference to "the consistency of the formatting of these dictionaries and the integrity of the tapes."

Neff et. al. found that parsing tended to fail with longer entries, which unfortunately tended to be the entries for higher-frequency words. "However, unparsable entries often have large sections of parsable material, which could be made available in LDB format for analysis or applications in spite of its partial nature, if only the top-down parser wouldn't fail." (LDB = lexical database, their term for a machine-readable lexicon for use in NLP.) Neff et. al. chose a top-down parser after also considering a bottom-up parser, but note that a bottom-up parser may be of use in cases where the top-down parser fails due to missing or corrupt font codes.

Some authors have considered the recovery of a hierarchical structure to be important, but others have disregarded it. For example, Schafer and Yarowsky [23] processed 80 bilingual dictionaries and produced translation pairs for machine translation; they specifically note that "No complex or hierarchical structure was assumed or used in our input dictionaries."

The view adopted here is that parsing should not necessarily be considered an essential step of dictionary processing. Because of the problem of irregularity in entry structure, parsing dictionaries poses some inherent difficulties. There are some NLP applications whose needs can be met without assigning a hierarchical structure to each dictionary entry.

Parsing is often characterized as "recovering" the hierarchical structure which gave rise to an observed sequence of tokens. I have chosen to use the word "assigning" rather

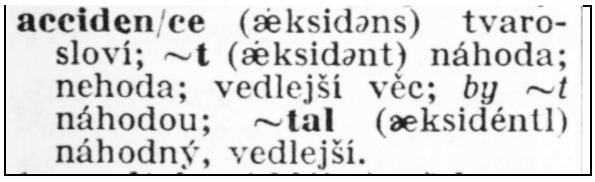

acciden|ce (æksidəns) tvarosloví; ~**t** (æksidənt) náhoda; nehoda; vedlejší věc; *by* ~*t* náhodou; ~**tal** (æksidéntl) náhodný, vedlejší.

Figure 75: Czech (001). Resolution of headword ellipsis.

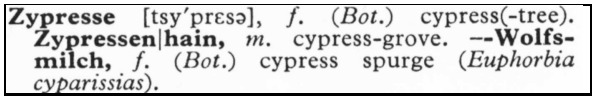

Zypresse [tsy′prɛsə], *f.* (*Bot.*) cypress(-tree). **Zypressen|hain,** *m.* cypress-grove. **--Wolfsmilch,** *f.* (*Bot.*) cypress spurge (*Euphorbia cyparissias*).

Figure 76: German (003). Resolution of parentheses indicating optionality.

than "recovering". The use of the term "recovering" would imply that there is an underlying truth to be recovered. As I discussed above, dictionary authors do not rigorously follow a well-defined grammar; there are cases where this would mean shoehorning multifarious types of lexical data into an ill-fitting constituency structure. Rather, dictionary entries are a mixture of semistructured formal data and arbitrary prose; the dictionary author can freely switch between the two and intermingle the two in whatever way best serves the needs of the human consumer of the dictionary. Parsing is sometimes useful, but my view is that it is less a matter of recovering an underlying truth, and more a matter of imposing an order for convenience of processing.

3.3.7 Spell-out

"Spell-out" is the name used here to refer to the conversion of the tagged, parsed representation into a form convenient for machine processing. Spell-out can potentially involve a profound transformation of the entry data.

The sort of processing involved in spell-out can be illustrated with reference to headword ellipsis, which was discussed in section 2. Consider the Czech entry in figure 75. One might choose to apply the following transformations (among others) to this entry during spell-out:

~t → **accident**
~tal → **accidental**

A similar sort of case is found in the German entry in figure 76. One might spell out "cypress(-tree)" as two alternative definitions, "cypress" and "cypress-tree".

Spell-out can involve adding explicit information which was merely implicit in the print dictionary. For example, Lemnitzer and Kunze [17] infer part of speech from gender information.

Addition of missing information has been done even in cases where the purpose of the project is enhanced searching of the human-readable dictionary. As an aid to searching, Fomin and Toner [7] (p. 84) add part-of-speech information which was not explicitly stated in the printed dictionary.

Spell-out might sometimes require human intervention. For example, one dictionary (Russian 025) explicitly indicates masculine gender on nouns ending with **-ь** , but omits this information in entries where the gender is made clear by the morphology on an adjective in an example within the same entry (e.g. бе́лый медве́дь). The gender might be obvious to the human user who is familiar with Russian morphology, but it presents a real challenge for an automated approach to spell-out. In some cases, human effort is likely to be required.

3.3.8 Validation

Validation is any automated means whose purpose is to confirm that the final product is free of errors.

It is unlikely that there can be any fully general approach to validation. The type of validation depends on the field type. Following are some suggested strategies:

- For each field, a set of permitted characters can be identified. In a Russian-English dictionary, Cyrillic characters might be prohibited in an English gloss field. (Fomin and Toner [7] p. 84 accomplished this by implementing an XML DTD which limits the range of permitted characters within particular elements).
- In the case where a field contains some type of abbreviation, such as morphosyntactic indicators or domain indicators, the abbreviation can be checked against a list of accepted abbreviations.
- In some cases, fields can be compared against external resources. A field of English glosses can be checked against a pre-existing English spell-check dictionary. If there is a corpus available for one of the languages, one can check for words in the appropriate dictionary fields which do not appear in the corpus; these words can be flagged for a human to check.
- Sequences of subentry numbers can be checked for invalid gaps (1 2 4 5).
- Brackets and parentheses must balance. For example, if an etymology field starts with an open square bracket, it must end with a close square bracket.

It can be expected that validation will reveal problems which must be corrected by hand. Fomin and Toner [7] experienced mistagging due to "frequently inconsistent and unpredictable use of italics in the dictionary." They state that "intensive editorial input" will be required to fix the problem.

3.4 Brief survey of previous work

The literature on dictionary parsing is too extensive to be fully covered here. The body of work is large, and investigation in this area has been going on for a long time; Wilks et. al. [29] (p. 750) cites interest in this area going back as far as 1979. The following review covers some of the highlights of this literature.

3.4.1 Applications

Investigators have had various applications for the data produced from human-readable dictionaries. These applications include:

- A pronunciation dictionary automated speech recognition (ASR) system (see, for example, the brief overview in Boguraev et. al. [4] p. 65)

- A table of bilingual glosses to use as a back-off strategy in cases where a word does not appear in the aligned bilingual text used to train a machine translation system [23] [8] [15].
- Resources for word sense disambiguation (see, for example, Chen and Chang [5] p. 63 for a review of the literature)
- Resources grouping words into semantic classes, such as verbs of perception, etc. (Gopestake et. al. [8] p. 185 give a brief review of the literature)
- A search system for the human-readable dictionary which permits searches to be restricted to specific fields ([7] p. 84)

Additional potential applications include:

- A pronunciation dictionary for a text-to-speech (TTS) system
- A table of inflected forms, together with morphosyntactic information, for use in part-of-speech tagging or syntactic parsing

3.4.2 *Approaches to processing*

Articles from the 1980s often do not describe the details of their processing strategies. Typical of the literature is the following quote:

> "A suite of programs to unscramble and restructure all of the fields in LDOCE entries has been written which is capable of decoding all the fields except those providing cross-reference and usage information for complete homographs." [3]

The article does not describe the strategies used in this suite of programs. Probably, in many cases in the work from this time period, the scripts were custom, monolithic, dictionary-specific scripts which contained hand-made rules. Neff and Borguraev [20] review the previous work in the area of dictionary processing, and note that conversion of an individual dictionary has often been accomplished with a "one-off" program.

The basic approach of one-off tools and hand-made rules continues to be used, even as more general dictionary processing tools have become available. For example, Fomin and Toner ([7] p. 84, pp. 88-9) use XSLT to transform presentation-level tags into TEI tags indicating the semantics of the fields. Although XSLT and TEI are relatively recent, their approach fundamentally amounts to the use of hand-written rules, as in the case where **ns.** and **gs.** (nominative singular, genitive singular) are identified by string matching and are tagged as <morphGroup>. Even in the present work, which is primarily concerned with CRFs, it was found that the etymology field in the Old English dictionary could most accurately be identified by using hand-made rules to identify the field and to mark its member tokens with an "etymology" attribute. This approach effectively spoon-feeds a predetermined result for this particular to the CRF. This is not a bad approach; the deterministic script does the best job at identifying this particular field, but all of the advantages of the CRF can be brought to bear on the other fields.

Neff et. al. [21] make one of the earlier attempts at creating a generalized tool for dictionary parsing. They deliberate between a top-down/depth-first parser and a bottom-up/all-paths parser. They end up picking the top-down parser, but keep open the possibility of using the bottom-up strategy "when it becomes necessary to process input with missing or corrupt font codes, there being few recovery strategies available to a top-down parser."

A somewhat later generalized dictionary processing system is the LexParse system [10] [13] [17]. The heart of LexParse consists of a tokenizer and a parser. The system is described as making use of context-sensitive type 1 grammars; the parser is depth-first, and handles alternatives by backtracking. Lemnitzer and Kunze note that LexParse offers multiple output options, including SGML, "Tree", and "Map".

Ma et. al [18] develop a major end-to-end system for the digitizing and processing of printed dictionaries. The system includes a customized OCR engine which is designed around the particular problem of recognizing formatted dictionary pages and entries. Their system includes a token tagger, much like the tagger described in section 1 of the present article; the major difference between the two approaches is that Ma et. al. make use of HMMs, while the present work makes use of CRFs.

4. CONCLUSIONS

Two dictionaries were tagged using CRFs. It was determined that there is only a slight gain in accuracy if the training set is expanded beyond around 3000 to 8000 tokens, where the specific beginning of the plateau depends on the complexity of the dictionary.

CRFs are useful for dictionary tagging, subject to certain limitations. Perfect accuracy is not to be expected with this approach, but the confusions tend to be limited to specific pairs of tags, and for some applications, those confusions may not be of concern. CRFs were observed not to perform as well on ranges of multifarious tokens, such as an etymology field; this was remedied with a preprocessing stage where the field was identified by a handmade rule and was marked with an attribute to help the CRF. However, CRFs are well suited for the basic problem, since they allow multiple attributes of the tokens to be taken into consideration.

It was argued that human-readable dictionaries, by their basic nature, cannot be expected to follow any fully regular form. Statistical approaches such as CRFs are appropriate for dictionary processing because they do not make overly rigid assumptions about the form of the entries.

5. ACKNOWLEDGMENTS

Grateful acknowledgment is given to Jason Eisner (Johns Hopkins University) and to Gregory Druck (University of Massachusetts Amherst) for discussion on Conditional Random Fields. Thanks to James LaBonte and William Reynolds for critical review.

Thanks is given to the authors of the Mallet program (primary author Andrew McCallum, with contributions from others) for making this extremely useful system generally available.

Acknowledgment is given to Dennis Dillahunt for his considerable help in the collection and preparation of materials for the survey of dictionaries.

6. APPENDIX: DICTIONARIES INCLUDED IN SURVEY

70 bilingual dictionary volumes were surveyed for this article. Following is an example of the citation form used to refer to the surveyed dictionaries:

Tibetan (024)

More specifically, the surveyed dictionaries are cited using the following two pieces of information:

1. The name of the more salient of the two languages. Nearly all of the surveyed dictionaries include English as one of the two languages; among those dictionaries, the non-English language is listed. Two dictionaries are bilingual between Modern German and a language other than English; in both cases, the language other than Modern German is listed.

2. An index number referring to full bibliographic information provided below.

In cases where a dictionary is bidirectional (e.g. English → Russian, Russian → English), the two sections of the dictionary are distinguished by appending an **a** or **b**, as in 019a.

There are numbering gaps at 017 and 042.

Number: 001
Title: English-Czech and Czech-English Dictionary
Alt-Title: Sloviník Anglicko-Český a Česko-Anglický
Author: Procházka, Jindřich
Edition: 16th edition
Year: 1959
Publisher: Artia

Number: 002
Title: Divry's Modern English-Greek and Greek-English Desk Dictionary
Alt-Title: Μεῖζον Νεώτερον Αγγλοελληνικον και Ελληνοαγγλικον Λεξικον
Author: Divry, George C., General Editor
Year: 1982
Publisher: D. C. Divry, Inc.

Number: 003
Title: Cassell's German-English English-German Dictionary
Alt-Title: Deutsch-Englisches Englisch-Deutsches Wörterbuch
Author: Betteridge, Harold T.
Year: 1978
Publisher: Macmillan

Number: 004
Title: A Comprehensive Dictionary English-Armenian
Alt-Title: ԸՆԴԱՐՁԱԿ ԲԱՌԱՐԱՆ
Author: Chakmakjian, H. H.
Year: 1978

Number: 005
Title: A Comprehensive Persian-English Dictionary
Author: Steingass, F.
Year: 1892
Publisher: Librairie Du Liban

Number: 006
Title: A Sanskrit-English Dictionary
Author: Monier-Williams, Sir Monier
Year: 1899
Publisher: Oxford University Press

Number: 007
Title: The Oxford English-Arabic Dictionary
Author: Doniach, N.S.
Year: 1972
Publisher: Oxford at the Clarendon Press

Number: 008
Title: A Dictionary of Urdū, Classical Hindi, and English
Author: Platts, John T.
Year: 1930
Publisher: Oxford University Press

Number: 009
Title: A Comprehensive Indonesian-English Dictionary
Author: Stevens, Alan M. and A. Ed. Schmidgall-Tellings
Year: 2004
Publisher: Ohio University Press

Number: 010
Title: New Complete Russian-English Dictionary
Alt-Title: Новый Полный Русско-Английский Словарь
Author: Segal, Louis
Year: 1942
Publisher: G. E. Stechert & Company

Number: 011
Title: A Short Bengali-English English-Bengali Dictionary
Author: Dabbs, Jack Autrey
Year: 1965

Number: 012
Title: Hawaiian Dictionary: Hawaiian-English English-Hawaiian
Author: Puku_, Mary Kawena and Samuel H. Elbert

Year: 1971
Publisher: University of Honolulu Press

Number: 013
Title: A Simplified Dictionary of Modern Samoan
Author: Allardice, R. W.
Year: 1985
Publisher: Polynesian Press

Number: 014
Title: Ukrainian-English
Author: Benyukk, Olesj and Raisa Galushko
Year: 1994
Publisher: Hippocrene Books

Number: 015
Title: Swahili-English Dictionary
Author: Rechenback, Charles W.
Year: 1967
Publisher: The Catholic University of Ameica Press

Number: 016
Title: Tagalog Dictionary
Author: Ramos, Teresita V.
Year: 1971
Publisher: University of Hawaii Press

Number: 018
Title: The Welsh Academy English-Welsh Dictionary
Author: Griffiths, Bruce and Dafydd Glyn Jones
Year: 1995
Publisher: University of Wales Press

Number: 019
Title: The Complete English-Hebrew Dictionary
Alt-Title: מילון אנגלי-בדי שלם
Author: Alcalay, Reuben
Year: 1970
Publisher: Massada Publishing Co.

Number: 020
Title: A Latin Dictionary
Author: Lewis, Charlton T. and Charles Short
Year: 1955
Publisher: Oxford at the Clarendon Press

Number: 021
Title: Hopi Dictionary
Alt-Title: Hopìikwa Lavàytutuveni
Year: 1998
Publisher: University of Arizona Press

Number: 022
Title: Cassell's English-Dutch Dutch-English Dictionary
Alt-Title: Engels-Nederlands Nederlands-Engels Woordenboek
Author: Coenders, H
Edition: 37th edition
Year: 1996
Publisher: Cassell

Number: 023
Title: A Comprehensive Swedish-English Dictionary
Alt-Title: Stora svensk-engelska ordboken
Year: 1988
Publisher: Esselte Studium

Number: 024
Title: English-Tibetan Dictionary of Modern Tibetan
Author: Goldstein, Melvyn C and Ngawangthondup Narkyid
Year: 1984
Publisher: Library of Tibetan Works and Archives

Number: 025
Title: The Oxford Russian Dictionary
Year: 1993
Publisher: Oxford University Press

Number: 026
Title: German-English Dictionary of Idioms
Author: Schemann, Hans and Paul Knight
Year: 1995
Publisher: Routledge

Number: 027
Title: Collins English-Italian Italian-English Dictionary
Year: 1995
Publisher: Harper Collins

Number: 028
Title: Chickasaw: An Analytical Dictionary
Author: Munro, Pamela and Catherine Willmond
Year: 1994
Publisher: University of Oklahoma Press

Number: 029
Title: Althochdeutsches Wörterbuch
Author: Schützeichel, Rudolf
Year: 1969
Publisher: Max Niemeyer Verlag

Number: 030
Title: Farsi-English English-Farsi (Persian)
Author: Miandji, A. M.

Year: 2003
Publisher: Hippocrene Books

Number: 031
Title: A Dictionary of Modern Written Arabic
Author: Cowen, J. Milton
Edition: 3rd edition
Year: 1976
Publisher: Spoken Language Services, Inc.

Number: 032
Title: Essential English-Vietnamese Dictionary
Alt-Title: Từ điển anh-việt
Author: Đình-hoà, Nguyễn
Year: 1983
Publisher: Charles E. Tuttle Co.

Number: 033
Title: The Oxford Hindi-English Dictionary
Author: McGregor, R. S.
Year: 1993
Publisher: Oxford University Press

Number: 034
Title: Mathews' Chinese-English Dictionary
Author: Mathews, R. H.
Edition: American Edition
Year: 1944
Publisher: Harvard University Press

Number: 035
Title: New Redhouse Turkish-English Dictionary
Alt-Title: Redhouse Yeni Türkçe-İngilizce Sözlük
Year: 1968
Publisher: Redhouse Press

Number: 036
Title: Luganda-English Dictionary
Author: Snoxall, R. A.
Year: 1967
Publisher: Oxford at the Clarendon Press

Number: 037
Title: Kikuyu-English Dictionary
Author: Benson, T. G.
Year: 1964
Publisher: Oxford at the Clarendon Press

Number: 038
Title: Bulgarian-English English-Bulgarian Dictionary
Author: Tchomakov, Ivan
Year: 1992
Publisher: Hippocrene Books

Number: 039
Title: The Oxford Dictionary of Modern Greek
Author: Pring, J. T.
Year: 1982
Publisher: Oxford University Press

Number: 040
Title: Mittelhochdeutsches Handwörterbuch
Author: Lexer, Matthias
Edition: Erster Band A-M
Year: 1872
Publisher: Verlag von S. Hirzel

Number: 041
Title: Türkisch-Arabisch-Persisches Handwörterbuch
Author: Zenker, Julius Theodor
Year: 1967
Publisher: Georg Olms Verlagsbuchhandlung

Number: 043
Title: The Navajo Language: A Grammar and Colloquial Dictionary
Author: Young, Robert W. and William Morgan
Year: 1980
Publisher: University of New Mexico Press

Number: 044
Title: The New World Comprehensive Korean-English Dictionary
Year: 1979
Publisher: Si-sa-yong-o-sa, Inc.

Number: 045
Title: The New World Comprehensive English-Korean Dictionary
Year: 1973
Publisher: Si-sa-yong-o-sa, Inc.

Number: 046
Title: Collins Spanish-English English-Spanish Dictionary
Author: Smith, Colin
Year: 1971
Publisher: Collins

Number: 047
Title: The New Chinese-English Dictionary
Alt-Title: 新汉英词典
Year: 1999
Publisher: Yilin Press

Number: 048
Title: Portuguese-English English-Portuguese
Author: Houaiss, Antônio and Ismael Cardim

Year: 1987
Publisher: Hippocrene Books

Number: 049
Title: A Dictionary of Japanese and English Idiomatic Equivalents
Author: Corwin, Charles et. al.
Year: 1968
Publisher: Kondasha International

Number: 050
Title: Dictionaire bilingue d'informatique: anglais-français français-anglais
Year: 1990
Publisher: Marabout

Number: 051
Title: The Bantam New College German and English Dictionary
Author: Traupman, John C.
Year: 1981
Publisher: Bantam Books

Number: 052
Title: Hungarian-English English-Hungarian
Author: Takács, G
Year: 1990
Publisher: Hippocrene Books

Number: 053
Title: Latvian-English English-Latvian Dictionary
Author: Sosāre, M and I. Borzvalka
Year: 1993
Publisher: Hippocrene Books

Number: 054
Title: Dansk-engelsk Ordbog
Author: Axelsen, Jens
Year: 1984
Publisher: Gyldendal

Number: 055
Title: Finnish-English English-Finnish Dictionary
Author: Wuolle, Aino
Year: 1990
Publisher: Hippocrene Books

Number: 056
Title: Slovak-English English-Slovak Dictionary
Author: Trnka, Nina
Year: 1992
Publisher: Hippocrene Books

Number: 057
Title: Norwegian English Dictionary
Alt-Title: Norsk Engelsk Ordbok
Author: Haugen, Einar
Year: 1965
Publisher: University of Wisconsin Press

Number: 058
Title: Estonian-English English Estonian Dictionary
Author: Kyiv, Ksana and Oleg Benyuch
Year: 1992
Publisher: Hippocrene Books

Number: 059
Title: English Serbo-Croatian Serbo-Croatian English
Year: 1971
Publisher: Langenscheidt

Number: 060
Title: English-Danish Danish-English Dictionary
Alt-Title: Engelsk-Dansk Dansk-Engelsk Ordbog
Year: 1979
Publisher: Berlitz

Number: 061
Title: Kenkyusha's New Japanese-English Dictionary
Author: Matsuda, Koh
Edition: 4th Edition
Year: 1974
Publisher: Kenkyusha

Number: 062
Title: A Chinese-English-French Fundamental Lexicon of Science and Technology
Alt-Title: 汉英法科技基础词汇
Year: 1994
Publisher: Sinolingua

Number: 063
Title: Sanseido's Daily Concise English-Japanese Dictionary
Alt-Title: デイリーコンサイス 英和大辞典
Edition: 4th Edition
Year: 1979
Publisher: Sanseido

Number: 064
Title: Dutch-English English-Dutch Dictionary
Year: 1995
Publisher: Hippocrene Books

Number: 065

Title: Langenscheidt's German-English English-German Dictionary
Alt-Title: Langenscheidts Deutsch-English English-Deutsch Wörterbuch
Author: Klatt, E. and G. Golze
Year: 1953
Publisher: Washington Squar Press

Number: 066
Title: Larousse's French-English English-French Dictionary
Alt-Title: Dictionnaire Français-Anglais Anglais-Français Larousse
Author: Dubois, Marguerite-Marie, Denis J. Keen, and Barbara Shuey
Year: 1955
Publisher: Pocket Books

Number: 067
Title: The Oxford Paperback Italian Dictionary
Author: Andrews, Joyce
Year: 1986
Publisher: Oxford University Press

Number: 068
Title: Dictionary of German Slang and Colloquial Expressions
Author: Strutz, Henry
Year: 2000
Publisher: Barron's

Number: 069
Title: Klett's Modern German and English Dictionary
Author: Weis, Erich
Year: 1984
Publisher: National Textbook Company

Number: 070
Title: A Concise Dictionary of Old Icelandic
Author: Zoëga, Geir T.
Year: 1910
Publisher: Oxford at the Clarendon Press

Number: 071
Title: Georgian-English English-Georgian Dictionary and Phrasebook
Author: Awde, Nicholas and Thea Khitarishvili
Year: 1996
Publisher: Hippocrene Books

Number: 072
Title: Diccionari Bàsic Català-Anglès Anglès-Català
Year: 1996
Publisher: Enciclopèdia Catalana

Number: 073 (not included in the survey of dictionaries, but cited for examples)
Title: Grammar of the Gothic Language
Author: Wright, Joseph
Year: 1910
Publisher: Oxford at the Clarendon Press

7. REFERENCES

[1] A. V. Aho, R. Sethi, and J. D. Ullman. *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, 1986.

[2] P. D. Allison. *Logistic regression using the SAS system: theory and application*. SAS Institute, Cary, N.C, 1999.

[3] H. Alshawi, B. Boguraev, and T. Briscoe. Towards a dictionary support environment for real time parsing. In *Proceedings of the second conference on European chapter of the Association for Computational Linguistics*, EACL '85, pages 171–178, Stroudsburg, PA, USA, 1985. Association for Computational Linguistics.

[4] B. Boguraev, D. Carter, and T. Briscoe. A multi-purpose interface to an on-line dictionary. In *Proceedings of the third conference on European chapter of the Association for Computational Linguistics*, EACL '87, pages 63–69, Stroudsburg, PA, USA, 1987. Association for Computational Linguistics.

[5] J. N. Chen and J. S. Chang. Topical clustering of mrd senses based on information retrieval techniques. *Comput. Linguist.*, 24:61–95, March 1998.

[6] M. Domenig and P. Shann. Towards a dedicated database management system for dictionaries. In *Proceedings of the 11th coference on Computational linguistics*, COLING '86, pages 91–96, Stroudsburg, PA, USA, 1986. Association for Computational Linguistics.

[7] M. Fomin and G. Toner. Digitizing a dictionary of medieval Irish: the eDIL project. *Literary and Linguistic Computing*, 21(1):83–90, April 2006.

[8] A. Gopestake, T. Briscoe, P. Vossen, A. Ageno, I. Castellon, F. Ribas, G. Rigau, H. Rodríguez, and A. Samiotou. Acquisition of lexical translation relations from MRDs. *Machine Translation*, 9:183–219, 1994. 10.1007/BF00980578.

[9] J. E. Grimes. Denormalization and cross referencing in theoretical lexicography. In *Proceedings of the 10th international conference on Computational linguistics*, COLING '84, pages 38–41, Stroudsburg, PA, USA, 1984. Association for Computational Linguistics.

[10] R. Hauser and A. Storrer. Dictionary entry parsing using the LexParse system. *Lexicographica*, 9:174–219, 1993.

[11] N. Ide, A. Kilgarriff, and L. Romary. A formal model of dictionary structure and content. In *Proceedings of Euralex 2000*, pages 113–126, 2000.

[12] D. Jurafsky and J. H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, Upper Saddle River, NJ,

2000.

[13] M. Kammerer. Wörterbuchparsing: Grundsätzliche Überlegungen und ein Kurzbericht über praktische Erfahrungen. 2000.

[14] B. Kang. Markup of Korean dictionary entries. *Language, Information, and Computation (PACLIC 11)*, pages 219–228, 1996.

[15] J. Klavans and E. Tzoukermann. The bicord system: combining lexical information from bilingual corpora and machine readable dictionaries. In *Proceedings of the 13th conference on Computational linguistics - Volume 3*, COLING '90, pages 174–179, Stroudsburg, PA, USA, 1990. Association for Computational Linguistics.

[16] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. ICML-01*, pages 282–289, 2001.

[17] L. Lemnitzer and C. Kunze. Dictionary entry parsing. *ESSLLI 2005*, 2005.

[18] H. Ma, B. Karagol-Ayan, D. Doermann, and J. Wang. Parsing and tagging of bilingual dictionaries. *TAL*, 44:125–149, 2003.

[19] C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA, 1999.

[20] M. S. Neff and B. K. Boguraev. Dictionaries, dictionary grammars and dictionary entry parsing. In *Proceedings of the 27th annual meeting on Association for Computational Linguistics*, ACL '89, pages 91–101, Stroudsburg, PA, USA, 1989. Association for Computational Linguistics.

[21] M. S. Neff, R. J. Byrd, and O. A. Rizk. Creating and querying lexical data bases. In *Proceedings of the second conference on Applied natural language processing*, ANLC '88, pages 84–92, Stroudsburg, PA, USA, 1988. Association for Computational Linguistics.

[22] J. Nyhan. Developing integrated editions of minority language dictionaries: The Irish example. *Literary and Linguistic Computing*, 23(1):3–12, 2008.

[23] C. Schafer and D. Yarowsky. Exploiting aggregate properties of bilingual dictionaries for distinguishing senses of English words and inducing English sense clusters. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, ACLdemo '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[24] C. Schneiker, D. Seipel, and W. Wegstein. Schema and variation: digitizing printed dictionaries. In *Proceedings of the Third Linguistic Annotation Workshop*, ACL-IJCNLP '09, pages 82–89, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[25] C. Schneiker, D. Seipel, W. Wegstein, and K. Prätor. Declarative parsing and annotation of electronic dictionaries. In *Proceedings of the 6th International Workshop on Language Processing and Cognitive Sciences (NLPCS)*, 2009.

[26] C. M. Sperberg-McQueen. *The TEI Consortium: guidelines for electronic text encoding and interchange*. Humanities Computing Unit, 2002, 2002.

[27] P. van der Eijk, L. Bloksma, and M. van der Kraan. Towards developing reusable NLP dictionaries. In *Proceedings of the 14th conference on Computational linguistics - Volume 1*, COLING '92, pages 53–59, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics.

[28] H. Wallach. Conditional random fields: An introduction. University of Pennsylvania CIS Technical Report MS-CIS-04-21., 2004.

[29] Y. Wilks, D. Fass, C.-m. Guo, J. E. Mcdonald, T. Plate, and B. M. Slator. Machine tractable dictionaries as tools and resources for natural language processing. In *Proceedings of the 12th conference on Computational linguistics - Volume 2*, COLING '88, pages 750–755, Stroudsburg, PA, USA, 1988. Association for Computational Linguistics.