Generalized Glossing Guidelines: An Explicit, Human- and Machine-Readable, Item-and-Process Convention for Morphological Annotation

David R. Mortensen[*†] **Ela Gulsen**[*†] **Taiqi He**[†]
Nathaniel Robinson[†] **Jonathan D. Amith**[‡] **Lindia Tjuatja**[†]
Lori Levin[†]
[†]Carnegie Mellon University [‡]Gettysburg College
[†]{dmortens, egulsen, taiqih, nrrobins, lindiat, levin}@andrew.cmu.edu
[‡]jonamith@gmail.com

Abstract

We introduce a YAML notation for multi-line interlinear glossed text (IGT) that represents non-concatentative processes such as infixation, reduplication, mutation, truncation, and tonal overwriting in a consistent, formally rigorous way, on par with affixation, using an Item-and-Process (IP) framework. Our new notation—Generalized Glossing Guidelines (GGG)—is human- and machine-readable and easy to edit with general purpose tools. A GGG representation has four fields: (1) A Surface Representation (`sr`) with curly brackets to show where non-concatenative morphological processes have applied. (2) A Lexical Representation (`lx`) that explicitly shows non-concatenative processes as insertions, deletions, and substitutions as they apply to the basic form of morphemes. (3) A gloss field (`gl`) that associates glosses with morphemes and morphological processes in the `sr` and `lx` lines. (4) A metalanguage translation. We demonstrate the linguistic adequacy of GGG and compare it to two other IGT annotation schemes.

1 Introduction

As part of the ongoing wav2gloss project, we are generating Interlinear Glossed Text (IGT) from speech using an end-to-end system. In producing IGT for various languages of the Americas, we encountered a challenge: traditional interlinear glossing schemes are well-suited for the representation of concatenative morphology (Comrie et al., 2008) where morphological properties are realized by spans of phonological material (Goodman et al., 2015; Maeda and Bird, 2000; Bird and Liberman, 1999; Bird et al., 2000; Ide and Suderman, 2007). However, the languages that we are working with—Mixtec, Nahuatl, and Totonac—are permeated by morphological operations such as truncation, tonal overwriting, reduplication, apophony, and segmental overwriting, that cannot be adequately expressed as the concatenation (or even interleaving) of strings. The shortcoming of most IGT notations is that they represent the alignment of affixes with glosses, but they do not explicitly show how non-concatenative processes align to glosses.

The contrast between concatenative and non-concatenative "models of grammatical description" goes back at least to a seminal article by Charles Hockett (1954) in which he observed that morphology can be viewed as the concatenation of morphemes (item-and-arrangement or IA) or as the application of processes to morphemes (item-and-process or IP). Whatever their ontological nature may be, some morphological operations—for example, apophony and truncation—are more easily expressed as processes than morphemes. In order to gloss these operations (and give them the same status as affixation), we needed to develop an annotation scheme more general than those currently available. Therefore, we propose Generalized Glossing Guidelines (or GGG), that build upon existing conventions such as the Leipzig Glossing Rules (Comrie et al., 2008) but make the framework formally explicit and add consistent and comprehensive support for non-concatenative morphological alternations such as infixation, reduplication, transfixation, apophony, tonal overwriting, and truncation.

Figure 1 gives an example of GGG from Yoloxóchitl Mixtec. It shows metadata as well as the four fields, `sr` (Surface Representation), `lx` (Lexical Representation), `gl` (gloss), and `tr` (translation). It shows tonal overwriting in curly brackets, with cliticization shown by `=`.

[*]Denotes equal contribution.

Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 58–67

2 Background

A large number of glossing conventions, from the very formal (e.g., Xigt; Goodman et al. 2015) to the relatively informal (e.g., the Leipzig Glossing Rules or LGR; Comrie et al. 2008) have been proposed and employed in computational applications. For example, a recent SIGMORPHON shared task on glossing used representations based on LGR.[1]

These conventions play two roles: (1) They allow linguists and language workers to communicate with one another with clarity and minimal ambiguity; (2) They allow humans and computers to communicate with one another with respect to the morphosyntax of human languages. In our use-case, they allow neural models to communicate the details of their morphosyntactic analyses to language workers. As such, these annotation conventions need to be both human readable (whether directly or through some kind of user interface) and expressive, without sacrificing explicitness.

Although LGR largely satisfies these critera when only concatenative morphology occurs, non-concatenative operations are only supported in a limited and sometimes inexplicit way in this convention. The following example shows the LGR notation for apophony (umlaut) in German:

(1) Ich habe vier Brüder
1.SG have.1.SG four brother\PL
'I have four brothers.'

The sequence "\PL" indicates that plural is marked by a non-concatenative process (in this case, apophony), but it does not index the morphological property to a specific formal change. In the Generalized Glossing Guidelines described here, the same example would be the following:[2]

(2) Ich habe vier Br{u>ü}der
1.SG have.1.SG four brother{PL}
'I have four brothers.'

LGR also has conventions for annotating reduplication and infixation, but each of these notations is different. Compare these examples from Motu:

(3) a. ma~mahuta
PL sleep
'to sleep'

b. {>ma}mahuta
sleep{PL}
'to sleep'

In LGR (3a), reduplicants are delimited with a tilde. In the GGG version (3b), again showing only `lx`, reduplication is notated with the same arrow notation as all other non-concatenative processes.

Compare the following well-known example of infixation in Tagalog:

(4) a. s<um>ulat
<COMPL>write
'write'

b. s{>um}ulat
write{COMPL}
'write'

In LGR (4a), infixes are surrounded by angle brackets. In the GGG version (4b), infixes are indicated with the same notation as reduplication and all other processes. Maximal empirical coverage is achieved with minimal formal equipment.

Another important framework for representing IGT (and morphosyntactic annotations, generally) is Xigt (Goodman et al., 2015), an XML-based format that associates annotations with spans. It, too, is highly general, machine-readable, and formally rigorous, but its opaque structure makes it difficult to read and write without special software tools.

We propose GGG to take the best of the both frameworks. It has the following properties:

- General and adaptable
- Human readable
- Machine readable and unambiguous
- Editable with general-purpose tools
- Consistent and formally-rigorous in its representation of non-concatenative processes

2.1 Lexical Representations

The core of the GGG format is the lexical or `lx` representation. To understand `lx`, one must distinguish morphological processes from phonological processes and imagine a pipeline in which morphological processes precede phonological processes.

Morphological processes are associated with meaning or grammatical features. For example, the Mixtec tone changes shown in Figure 1 mark the habitual aspect. Phonological processes, in contrast, are not associated with meanings. They are processes that apply when phonological conditions are met. For example, tone sandhi in many languages is purely phonological (does not realize any morphosyntactic properties).

[1] https://github.com/sigmorphon/2023GlossingST

[2] We show only `lx` here, structuring fields as in a conventional glossed example, and omit `sr` for the sake of comparison to LGR.

In Item-and-Process Morphology, there are two kinds of constructs associated with meaning: morphemes (items) and processes. The pipeline assumed by GGG is one in which morphemes are first assembled via concatenation (a MORPHEMIC REPRESENTATION). At this level, each instance of the same morpheme has the same form (except in cases of suppletion). Then, processes apply to these strings. Together, the items and processes form the lexical representation (`lx`) in GGG. This representation is the output of the morphology and the input to the phonology.[3]

Phonological rules may apply to the `lx` representation, yielding phonologically conditioned allomorphy. Some cases of nasalization shown in the `sr` field in Figure 1 are phonological. Since nasalization is not associated with any meaning, it does not correspond to labels in the gloss (`gl`).

In GGG, the `lx` represents the application of processes to morphemes—mapping between a MORPHEMIC REPRESENTATION and an UNDERLYING REPRESENTATION. The bracket-and-arrow notation shown in (3b) and (4b) above describes rewrites between the morphemic form and the underlying form. That is to say, the morphemic representation is everything outside of the brackets interspersed with everything to the left of the arrows (>) and the underlying representation is everything outside of the brackets interspersed with everything to the right of the arrows. The surface representation, in contrast, is the output of the phonology.

2.2 GGG is purely descriptive

The goal of GGG is **not** to provide a deep theoretical account of morphology but rather to be purely descriptive. Thus—for example—even when we believe that a morphological process is best explained by autosegmental tones being "bumped" from one mora to the following mora, GGG represents this process as the deletion of a tone from one mora of the morphemic representation and the simultaneous insertion of an identical tone on the following mora in the underlying representation (with some loss of generality). This is done to explicitly state the formal relationship between a morphemic form and underlying form while making a minimum of theory-internal assumptions. For example, in Yoloxóchitl Mixtec, the habitual is formed by overwriting a /4/ (high) tone to the first mora. Two examples are given in (5):

(5) a. chio'{1>4}o{>1}4
cook_boiling{HAB;1,2}
habitually cook by boiling'

b. sa{3>4}ta{>3}4
sa{3>4}ta{>2}4
buy{HAB;1,2}
'habitually buy'

Note that these changes are morphologically (not phonologically) conditioned. In (5a), GGG represents the tonal morphology as /1/ being replaced by /4/ and (the second) /4/ being preceded by an inserted /1/, focusing on the superficial (insertion of /1/ in the second mora) rather than the deep relationship (reassingment of the same /1/ to the second mora) between the morphemic representation and the underlying representation (the input to the phonological rules).

3 The Guidelines

GGG attempts to represent IGT examples like those in the preceding section in a YAML format, [4] preserving to the degree possible the conventions that are present when linguists typeset linguistic data for the consumption of other linguists. This allies it with the SIL Shoebox format and differentiates it from Xigt (Goodman et al., 2015) and other highly explicit IGT formats. This also makes it relatively easy to edit GGG text using off-the-shelf tools (e.g., text editors and transcription tools).

3.1 General Data Structure

An illustration of a YAML file for GGG is presented in Figure 1. The top level object is a map, consisting of metadata fields (`obj_lang` for "object language" and `meta_lang` for "meta language" are required), and `segs`, which is an array of "discourse segments" (roughly, sentences). The field `obj_lang` consists of a single ISO 639-3 code (as a string). The field `meta_lang` is an array of ISO 639-3 codes. Each discourse segment is a map with the following fields:

src The audio or video document from which the segment derives.

start The start time of the interval in the source file from which the segment derives (in seconds since the beginning of the recording).

[3] Note that this approach assumes a non-trivial and controversial assumption about the phonology-morphology interface. It excludes interleaving between morphological and phonological alternations. This is done to make the glossing format tractable and is characteristic of glossing formats generally. However, when cyclic phonology results in a two-step change, GGG allows this to be represented.

[4] `https://yaml.org`

```
obj_lang: xty
meta_lang: eng
segs:
-
src: xty0002.wav
start: 256
end: 265
speaker: 3
lx: "ja'{3>4}nda2 =nã1 =e1 ka4 nda{3>4}sa3 ba'1a3 =na2 yu'3u4 =run4"
sr: "ja'{4}nda2 =nã1 =e1 kã4 nda{4}sa3 ba'1a3 =nã2 yu'3u4 =run4"
gl: "cut{HAB} =3.PL =3.INAM there convert{HAB} good =3.PL mouth =wood"
tr: "...they cut it and convert it into a bifurcated stick."
```

Figure 1: Sample of GGG from Yoloxóchit Mixtec showing the use of bracket-and-arrow notation to indicate tonal overwriting and differences between lexical and surface forms produced by phonological rules. The numerals after vowels represent tones (/4/ is high; /1/ is low) associated with the preceding mora (for our purposes, vowel).

end The end time of the interval in the source file from which the segment derives (in seconds since the beginning of the recording).

speaker ID for speaker in this discourse segment.

lx The lexical representation of the discourse segment—the mapping between a MORPHEMIC representation in which all morphemes are represented in their canonical form (to which all processes have applied) and the underlying form that is the input to the phonology; consists of tokens (corresponding to morphemes) delimited by spaces.

sr The surface representation of the discourse segment—the output of the phonology, consisting of tokens delimited by spaces.

gl The glosses of each of the tokens in the lx and `sr` strings, delimited by spaces.

tr An idiomatic translation of the discourse segment (as a string).

Crucially, when split on white space, the `lx`, `sr`, and `gl` fields must consist of exactly the same number of strings. An alternative and equivalent representation would be to have these fields be arrays of objects, each corresponding to a word. This would enforce the alignment between words and glosses directly. However, it is much less readable than the proposed format and would be harder to edit with off-the-shelf tools.

Each of the tokens in the `lx` and `sr` strings consists of either a root, affix, or clitic and one or more processes that have been applied to it, as described in §3.2. Each of the tokens in the `gloss` string also consist of roots, affixes, clitics, and processes. Each word must have the same number of each of these categories of items. Except for processes, these must occur in the same order in forms and glosses. The roots, affixes, and clitics that make up the words are "morpheme-like units" (or tokens) and are delimited by spaces. Each process is associated with a single morpheme-like unit.[5]

3.2 Space-Delimited Form Tokens

Form tokens are sequences with components of the types shown in Table 1.

TYPE	CONN.	PREC. BASE?	EXAMPLE	GLOSS
root	n/a	n/a	Kind	child
prefix	-	Y	un- likely	NEG- likely
suffix	-	N	Kind -er	child -PL
procliti	=	Y	j'= aime	1.SG= like
enclitic	=	N	child ='s	child =POSS

Table 1: Types of tokens.

When lexical glosses consist of multiple words, they are joined with the underscore, as in Hmong `lug` '`come_back`'. In this case, an optional rule from LGR is made mandatory. The use of a period to compose complex glosses is not to be used for this purpose. Instead, it is used strictly in cases of cumulative exponence (that is, where a single morpheme realizes and is glossed with more than one property) as in English `-s` '-3.SG.PRS'.

[5] In a few cases, this has proven problematic and has resulted in redundancy, but in the general case, it has worked well.

Form tokens may contain annotations for MORPHOLOGICAL PROCESSES such as the following:

- Reduplication
- Infixation
- Transfixation
- Apophony
- Tonal overwriting
- Segmental overwriting

These are indicated with bracketed expressions. In lexical forms (`lx`), these consist of `{A>B}` where `A` and `B` can be any string including the empty string. These indicate a process in which `A` has been replaced by `B`. Examples include English `t{u>i}θ` 'tooth{PL}.' In `srs`, these consist of `{A}`, where `A` can be any string (including the empty string). These indicate substrings that are the result of the application of a process. Take, for example, English `t{i}θ` 'tooth{PL}'. For a complete example, see Figure 1. In some cases, there may be a hierarchical relationship between processes, where one process "feeds" another. This is indicated by providing additional steps using the bracket-and-arrow notation, e.g., `{3>1>4}` as in the following examples from Yoloxóchitl Mixtec. In (6a) and (6b) the irrealis transitive *ta'3bi4* and instransitive *ta'1bi4* are changed to the habitual, with tone /4/ on the first mora. We analyze the shift of /3/>/1/ as a detransitivising process and thus in example (6b) both DTR and HAB are represented by {3>1>4}. The low tone /1/ is then reassigned to the second mora (shown in GGG as the "insertion" of /1/ on /i/). In many cases this "push" of first mora's original tone (/1/ or /3/) onto the second mora occurs, forming a contour tones (e.g., /14/ and underlying /34/ (surface /24/ by phonological rule after the mora-initial tone 4 of the habitual).

(6) a. ta'{3>4}bi4
break{HAB}
'habitually break (transitive)'

b. ta'{3>1>4}bi{>1}4
break{DTR.HAB;1,2}
'habitually break (intransitive)'

3.3 Covert elements

When the absence of an affix is significant, it can be represented as 0- or -0 (standing in for ∅ or ϵ).

3.4 Distinguishing Morphology from Phonology

The process notations are not meant to represent purely phonological alternations. If an alternation can be accounted for by a rule that is wholly conditioned by the surrounding phonological segments or syllable structure and prosodic context, it should be treated as phonological and not directly represented in the `lx` field. The `lx` field should contain only information that is derivable from the lexical, derivational, and inflectional properties of a token and is not predictable on another basis.

3.5 Space-Delimited Gloss Tokens

Type	Example	Glosss
Infixation	s{>um}ulat	write{PFV}
Reduplication	{>su}sulat	write{PROSP}
Transfixation	k{i>u}t{a>u}b	book{PL;1,2}
Apophony	t{u>i}θ	tooth{PL}
Segmental overwriting	{xi>ku}3xi3	eat{IRR}
Tonal overwriting	ku{3>14}ni2	want{NEG}

Table 2: Example forms and glosses for a range of morphological processes.

Conventions for associating gloss tokens with morpheme tokens (see Table 2) are based on the Leipzig glossing conventions with significant extensions. When possible, labels for categories are derived from the Unimorph schema (Sylak-Glassman, 2016).

Each gloss token consists of a lexical or morpheme gloss followed by a sequence of process glosses (each enclosed in curly brackets) and zero or one delimiters {=, -} which may be either preposed or postposed. Process glosses consist of lexical glosses or morpheme glosses and an optional semicolon followed by a list of numbers separated by commas. The numbers indicate the index of spans (starting from 1) in the corresponding form the gloss applies to. For example, in Arabic `k{i>u}t{a:>u}b` 'book{PL;1,2}', the PL property is realized by two changes ({>u} and {>u}) and this is indicated by the span indices (1,2) after the semicolon. For ease of annotation, if there is only one process in a word, the index can be omitted.

Some form tokens have more than one associated process. The corresponding glosses are provided in successive bracketed expressions after that lexical or morpheme gloss. For example, in Arabic `k{>a}t{>:}{>a}b{>a}`

'write{PST;1,3}{CAUS;2}{3.SG.M;4}', there are three processes, indicated by the three properties in brackets with their respective indices. The use of indices means the alignment between bracketed expressions in forms and glosses is deterministic. The orders of the processes (bracketed expressions) in the gloss can be arbitrary, but—as a group—they should appear only at the end of the gloss.

Morpheme glosses are drawn from the Unimorph Schema (Sylak-Glassman, 2016) when possible.[6] When glosses for derivational morphology are present in the Leipzig Rules but not in Unimorph, the Leipzig gloss should be used. When a needed category is not represented in either resource, it will be added to the standard.

Super-categories of features are representated as `CATEGORY::`. Thus, first-person plural subject is represented as `SUBJ::1.PL`.

3.6 Disjunctions

Disjunctions between properties can indicated with the pipe (`|`) operator and grouping can be indicated with square brackets. The `|` operator binds more closely than the `.` operator. Thus, English *you* may be glossed (out of context) as `2.SG|PL.NOM|ACC` (second person, singular or plural and nominative or accusative. Square brackets can be used for grouping. German *sie* can be glossed (out of context) as `3.[SG.FEM]|[PL.NOM|ACC]` (third person, either feminine singular or unspecified for gender and plural and either nominative or accusative). Disjunctions are to be used when the exact analysis of a wordform, in context, is not clear to an annotator. In general, their use should be minimized as the quality of the annotations improves.

3.7 Translations

Each discourse segment should be accompanied by an idiomatic translation into the metalanguage.

3.8 Parsing GGG

Parsing GGG is more complicated than parsing Xigt because GGG is, effectively, an $A^nB^nC^n$ language. To validate or parse GGG, one must ensure that three sequences, `lx`, `sr`, and `gl`, are the same length (when split into tokens on white space). This means that context-free parsing for GGG is not possible. This adds some overhead to writing tools for GGG. However, we have written parsing, generation, and validation tools for GGG without excessive investments.[7]

4 Linguistic Adequacy

The adequacy of GGG for annotating concatenative morphology is identical to that of LGR, since the mechanism is borrowed from LGR directly. The only modification is that morphemes within a word are divided by spaces in addition to hyphens and equal signs. This means that the headedness of compounds must be stated explicitly (with dependents treated like affixes).

The GGG approach, however, has a distinct advantage in the treatment of non-concatenative morphology, as it is able to achieve complete adequacy (though not theoretical correctness or depth of generalization) through the use of a single annotation mechanism: {A?>B?(;C)?}. We show that the convention works well for infixation, reduplication, truncation, apophony, tonal overwriting, segmental overwriting, transfixation, and other similar processes.

4.1 Infixation

Infixation involves the inserting of a morpheme into a morpheme. Take the following examples from Ulwa, a Misumalpan language of Nicaragua. Possessives are denoted by affixes such as "ka" (3.SG) and "ki" (1.SG), which may occur as either suffixes or infixes depending on the syllable structure of the word. Therefore, in all of these cases, we are treating the affixes as morphological processes. McCarthy and Prince (1993)

```
-
  lx: "wahai{>ki}"
  sr: "wahai{ki}"
  gl: "brother{POSS::1.SG}"
  tr: "my brother"
-
  lx: "sû{>ki}lu"
  sr: "sû{ki}lu"
  gl: "dog{POSS::1.SG}"
  tr: "my dog"
```

Using LGR, the first two URs would be annotated as `wahai<ki>` and `sû<ki>lu`. Consider a similar example from Latin:

[6] See, also https://unimorph.github.io/schema/

[7] See https://github.com/cmu-llab/generalized-glossing-guidelines.

Operation	GGG	LGR	Xigt
prefix	un- likely NEG- likely	un-likely NEG-likely	✓
suffix	Kind -er child -PL	Kind-er child-PL	✓
infix	sû{>ki}lu dog{1.SG}	sû<ki>lu dog<1.SG>	✓
prefixing reduplication	{>su}sulat write{PROSP}	su~sulat PROSP~write	✓
infixing reduplication	ma{>m}vi̱t lion{PL}	?	✓
suffixing reduplication	kuk{>uk} bark{PROG}	kuk~uk bark~PROG	✓
subtractive morphology	nyoo{n>} lamb{PL}	✗	✗
apophony	c{ea>i}nn head{PL}	cinn head\PL	✗
tonal overwriting	xi{3>4}xi3 eat{HAB}	✗	✗
segmental overwriting	{ki>ka}3{xa>sa}3 do{IRR; 1,2}	✗	✗
transfixation	k{i>u}t{a:>u}b book{PL;1,2}	✗	✓
score	11	6.5	7

Table 3: Comparison of the representation of different morphological processes by glossing convention.

```
-
  lx: "ta{>n}g{>o}"
  sr: "ta{n}g{o}"
  gl: "touch{1.SG.PRS.IND}"
  tr: "I touch."
```

Both of these systems are equally adequate for representing infixation (at least of this kind). Infixing reduplication, however, is possibly a different matter, as shown in §4.2 below.

4.2 Reduplication

Reduplication refers to the realization of a morphological property by repeating material from a base. In this example from Mangap-Mbula, a VC-sequence is reduplicated after the base, to mark progressive aspect: (Bugenhagen, 1995)

```
-
  lx: "kuk{>uk}"
  sr: "kuk{uk}"
  gl: "bark{PROG}"
  tr: "be barking"
```

GGG can deal with relatively complex types of redupulication such as occur in Balsas Nahuatl[8], in which the repeated material can ultimately be realized as a high tone and/or a lengthened vowel (which are not necessarily contiguous):

```
-
  lx: "ti- ne:{>ó}ch- {>te}te:mowa -0"
  sr: "ti- ne:{ó}x- {te}te:mowa -0"
  gl: "SUBJ::2SG- OBJ::1SG- \
       {RED_H;1,2}look\_for -PRS.IND.SG"
  tr: "You look for me."
-
  lx: "ni- mi{>:ó}ts- te:mowa -0"
  sr: "ni- mi{:ó}s- te:mowa -0"
  gl: "SUBJ::1SG- OBJ::2SG- \
```

[8]The acute accent indicates a high tone. Unlike other varieties of Nahuatl, Balsas Nahuatl is tonal (Guion and Amith, 2005; Guion et al., 2010).

```
      {RED_H;1}look\_for -PRS.IND.SG"
  tr: "I look for you."
```

GGG is uniquely able to formalize Balsas Nahuatl reduplication with a fixed coda laryngeal (RDP_H), a reduplicant that can be realized on the the stem in various ways (first, third, and fourth examples) or on a prefix (second example). The commonality of all four cases is established by the common gloss: (RDP_H). Reduplication may be prefixing, suffixing, or infixing. The case of infixing reduplication is particularly problematic for LGR, since it is not clear which convention—the tilde convention for reduplication or the angle-bracket notation for infixation—should take precedence. In GGG, the notation is the same and this decision is not necessary. Take the following example from Pima (Riggle, 2006):

```
-
  lx: "ma{>m}viʈ"
  sr: "ma{m}viʈ"
  gl: "lion{PL}"
  tr: "lions"
-
  lx: "tʃi{>tʃ}maiʈ"
  sr: "tʃi{tʃ}maiʈ"
  gl: "drum{PL}"
  tr: "drums"
```

A similar pattern of infixing reduplication can be found in Latin:

```
-
  ur: "s{>po}pond{>ī}"
  sr: "s{po}pond{ī}"
  gl: "perform{1.SG.PRF.IND;1,2}
```

4.3 Subtractive morphology

Subtractive morphology involves the deletion of a segmental material from a base. The Murle language in the Surmic family subtracts the last consonant of a noun to change it from singular to plural: (Arensen, 1982)

```
  lx: "nyoo{n>0}"
  sr: "nyoo{}"
  gl: "lamb{PL}"
  tr: "lambs"
-
  lx: "wawo{c>0}"
  sr: "wawo{}"
  gl: "white_heron{PL}"
  tr: "white herons"
```

There appears to be no standard way of notating this in LGR. In Xigt, we believe that subtractive morphology could be notated by aligning a gloss with an empty string, but this would make it indistinguishable from realizing a morphological property via no change to the form.

4.4 Apophony

Apophony refers to a process in which a morphological property is realized through an alternation in phonemes. Take the following examples from Irish, in which vowel alternation is used to turn singular nouns into plural (Fife and King, 2017).

```
-
  lx: "c{ea>i}nn"
  sr: "c{i}nn"
  gl: "head{PL}"
  tr: "heads"
-
  lx: "m{ui>a}r{>a}"
  sr: "m{a}r{a}"
  gl: "sea{PL;1,2}"
  tr: "seas"
```

Apophony in Totonac often involves consonants changes, like changing /ʃ/ to /s/:

```
-
  lx: "{ʃ>s}kú'ta'"
  sr: "{s}kú'ta'"
  gl: "sour{DIM}"
  tr: "a little sour"
-
  lx: "{ʃ>s}u:ni'"
  sr: "{s}u:ni'"
  gl: "bitter{DIM}"
  tr: "a little bitter"
```

LGR allows one to indicate that apophonony affects a morpheme, but does not apply a notation for specifying its locus. Apparently Xigt has no way to distinguish apophony from infixation.

4.5 Tonal overwriting

Tonal overwriting refers to the class of morphological processes in which a tonal "affix" overwrites the existing tonal melody on a base. Examples from Yoloxóchitl Mixtec—which uses tonal overwriting to indicate different verbal inflections, such as habitual and negative—follow:

```
-
  lx: "ta'{3>1>4}bi{>1}4"
```

```
  sr: "ta'{4}bi{14}"
  gl: "get-broken{HAB;1,2}"
  tr: "habitually get broken"
```

In Xigt, there is not a clear way of distinguishing these changes from infixation. In LGR, these can be represented with the backslash notation used for apophony, with the same drawbacks.

4.6 Segmental overwriting

Tonal overwriting is fairly common. The analogous segmental process—in which a string of segments is overwritten by other segments—is relatively rare, but does exist. The following example from Yoloxóchitl Mixtec employs segmental overwriting to inflect a class of verbs as irrealis:

```
-
  lx: "{xi>ku}3xi3"
  sr: "{ku}3xi3"
  gl: "eat{IRR}"
  tr: "eat"
```

4.7 Transfixation

Transfixation involves interspersing affixal spans into a root morpheme. In Semitic languages such as Arabic and Hebrew, words are mostly associated with 3-consonant roots. In Arabic, *k-t-b* is a root meaning "write" and *d-r-s* is a root meaning "study". These roots are combined with patterns of vowels to form words.

Transfixation is particularly tricky to represent using LGR, and it is unclear which convention should be used to do so (the angle-bracket infix notation or the backslash non-concatenative notation). In GGG, all of the patterns inserted into the root are treated as morphological processes, using the bracket notation.

Take the following examples from Arabic, which show how different vowel patterns can distinguish between singular and plural nouns, as well as different forms of verbs.

```
  lx: "q{a>u}l{>uu}b"
  sr: "q{u}l{uu}b"
  gl: "heart{PL;1,2}"
  tr: "hearts"
-
  lx: "d{>a}r{>a}s{>a}"
  sr: "d{a}r{a}s{a}"
  gl: "study{PST;1,2}{3.SG.M;3}"
  tr: "he studied"
```

Transfixation can be combined with other processes as well. For example, gemination on the 2nd consonant of the root is used to turn a Form I verb into a causative Form II verb (Haspelmath and Sims, 2010).

```
-
  lx: "d{>a}r{>:}{>a}s{>a}"
  sr: "d{a}r{:}{a}s{a}"
  gl: "study{PST;1,3}{CAUS;2}{3.SG.M;4}"
  tr: "he taught"
```

A scorecard comparing the adequacy of GGG, LGR, and Xigt is shown in Table 3.

5 Conclusions

As should be clear from Table 1, most of the attested types of morphological processes can be represented in all three annotation formats. However, GGG has clear advantages in some areas. For example, if a linguist wants to know how nouns with a particular singular form are realized in the plural, without knowing in advance what processes are involved, they could discover this through relatively simple processing of GGG—because it is completely explicit. It would be immediately evident whether the process was a particular kind of apophony, reduplication, tonal overwriting, etc. For the other two annotation formats, this kind of research—if non-concatenative processes are involved—is considerably more complicated.

One cost, because of its explicitness, is that GGG annotation cannot be completed until a linguist has a thorough (though fundamental) analysis of a language's morphology. Our goal is to develop tools to facilitate this analysis: to go from basic recordings to interlinear annotations with reduced human intervention. We hope that GGG will be an important part of this ongoing work. But the benefits are great. We are currently using GGG with great success in our ongoing research and hope that other investigators will find it similarly useful.

Acknowledgments

We gratefully acknowledge the support of US National Science Foundation, grant number 2211951, numerous examples Mixtec examples from Rey Castillo Garcia, and generous contributions from three anonymous reviewers.

References

Jonathan E. Arensen. 1982. *Murle grammar*, volume 2 of *Occasional Papers in the Study of Sudanese Languages*. Summer Institute of Linguistics and University of Juba, Juba, Sudan.

Steven Bird, David S. Day, John S. Garofolo, John Henderson, Christophe Laprun, and Mark Y. Liberman. 2000. Atlas: A flexible and extensible architecture for linguistic annotation. *ArXiv*, cs.CL/0007022.

Steven Bird and Mark Y. Liberman. 1999. A formal framework for linguistic annotation. *ArXiv*, cs.CL/9903003.

R.D. Bugenhagen. 1995. *A Grammar of Mangap-Mbula: An Austronesian Language of Papua New Guinea*. Books Series. Department of Linguistics, Research School of Pacific and Asian Studies, Australian National University.

Bernard Comrie, Martin Haspelmath, and Balthasar Bickel. 2008. The leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses. *Department of Linguistics of the Max Planck Institute for Evolutionary Anthropology & the Department of Linguistics of the University of Leipzig. Retrieved January*, 28:2010.

James Fife and Gareth King. 2017. *Celtic (Indo-European)*, chapter 24. John Wiley & Sons, Ltd.

Michael Wayne Goodman, Joshua Crowgey, Fei Xia, and Emily M. Bender. 2015. Xigt: extensible interlinear glossed text for natural language processing. *Language Resources and Evaluation*, 49(2):455–485.

Susan G Guion and Jonathan D Amith. 2005. The effect of [h] on tonal development in nahuatl. *The Journal of the Acoustical Society of America*, 117(4):2490–2490.

Susan G Guion, Jonathan D Amith, Christopher S Doty, and Irina A Shport. 2010. Word-level prosody in balsas nahuatl: The origin, development, and acoustic correlates of tone in a stress accent language. *Journal of Phonetics*, 38(2):137–166.

M. Haspelmath and A.D. Sims. 2010. *Understanding Morphology*. Understanding language series. Hodder Education.

Charles Francis Hockett. 1954. Two models of grammatical description. *WORD*, 10:210–234.

Nancy Ide and Keith Suderman. 2007. Graf: A graph-based format for linguistic annotations. In *LAW@ACL*.

Kazuaki Maeda and Steven Bird. 2000. A formal framework for interlinear text. Paper presented at the workshop on Web-Based Language Documentation and Description.

John J McCarthy and Alan Prince. 1993. *Prosodic Morphology: Constraint Interaction and Satisfaction*. Linguistics Department Faculty Publication Series. 14. University of Massachusettes Amherst.

Jason Riggle. 2006. Infixing reduplication in pima and its theoretical consequences. *Natural Language & Linguistic Theory*, pages 857–891.

John Sylak-Glassman. 2016. The composition and use of the universal morphological feature schema (unimorph schema). Ms., Center for Language and Speech Processing, Johns Hopkins University.