Allomorphy

David R. Mortensen

February 3, 2025

Morphemes Can Have More than One Realization

In English, there is a morpheme HOP, meaning, roughly, 'jump like a rabbit'. There are two ways of spelling hop:

(1) a. Peter was afraid to **hop** past Mr. McGregor's gate.
    b. Peter **hop**s past Mr. MacGregor's gate.
    c. Peter **hopp**ed past Mr. MacGregor's gate.
    d. Peter is **hopp**ing past Mr. MacGregor's gate.

When HOP occurs before the past tense suffix *-ed* or the present participle suffix *-ing* it is realized as *hopp*, but it is realized as *hop* elsewhere.

This is an example of what is called allomorphy—the state of affairs when a single morpheme has more than one realization. The realization can be in terms of spelling. It can also be in terms of pronunciation or (in a sign language) in the motor or visual representation of the morpheme.

The basic definition of allomorphy:

(2) a. The signified remains constant
    b. The signifier varies
    c. The distribution of the various signifiers is predictable (follows a rule)

The various signifiers are called ALLOMORPHS. The *hopp* allomorph occurs before a suffix that starts with a vowel . The *hop* allomorph occurs in all other situations (elsewhere) .

This is true of spelling (orthography) but not pronunciation. *hopped* is pronounced /hɑpt/, with no vowel before the suffix.

This is an application of a principle, first known from the Sanskrit grammarian Pāṇini, which is sometimes called the ELSEWHERE PRINCIPLE. It holds that grammatical rules act like `case` statements in programming languages, with the most specific cases given priority, more general cases following, and a fallback case (the elsewhere case) applying when none of the other cases do.

Phonologically-Conditioned Allomorphy

The best studied type of allomorphy is that in which the ENVIRONMENTS[1] that determine which allomorph will surface are based on sound. Since we have not dealt with phonology yet, we will use spelling as a proxy for pronunciation (even though these two things—phonology and orthography—are rather different beasts).

[1] Linguists call general contexts ENVIRONMENTS.

Examples from English

A classic example of allomorphy is presented by the English plural suffix:

Sometimes the plural is written as *-s*. More rarely, it is written as *-es*. Whether one writes *-s* or *-es* is perfectly predictable: When the plural suffix follows *ch*, SH, *s*, or *z*, one writes *-es*. Otherwise, one writes *-s*, as seen in Table 1.

kit	kits	kiss	kisses
kid	kids	buzz	buzzes
pick	picks	pitch	pitches
bud	buds	bus	buses
puff	puffs		
fin	fins		
jam	jams		
path	paths		
pill	pills		
fear	fears		

Table 1: Orthographic allomorphs of the English plural suffix.

One can express this pattern of allomorphy as a rule:

1. Start by adding the suffix *s*. We'll call *s* the UNDERLYING FORM of the morpheme.
2. Apply a rule that adds an *e* between any sequence of *s*, *z*, *sh*, or *ch* and an *s* (at the end of a word).

For those who are familiar with Python regular expression syntax, the rewrite rule could be expressed with the following function:

```python
import re

def e_insertion(form: str) -> str:
    return re.sub("(ch|sh|s|z|)(s)$", "\1e\2", form)
```

Figure 1: A Python implementation of the English e-insertion rule.

Applying this to the underlying form of the word (the concatenation of the underlying forms of both morphemes) sometimes results in a change and sometimes does not. When there is a change, our plural morpheme looks like *-es* (or, at least, we have an *e* before the final *s*).

The same function can be applied to all words. It will only insert *e* in cases where the *-es* allomrph is expected. Or almost.

What about words like *pass*? If we pass *pass* to our `e_insertion` function, we get *pases*, which is not what we want. We need some way of saying that our rule only applies at morpheme boundaries (not in the middle of morphemes like *pass*). Let us say that rather than concatenating the underlying forms of morphemes to get the underlying form of the word that we join them with `^`, indicating a morpheme boundary, and that we delete all of the `^` symbols when we are done with them. We can then revise our function to be:

This rule will apply to *pass^s* (yielding *pass^es*), but will not apply to *pass*. It raises an important point: some rules of allomorphy are sensitive to morphological structure. Indeed, some apply only to certain morphemes.

```python
import re

def e_insertion(form: str) -> str:
    return re.sub("(ch|sh|s|z|)[^](s)$", "\1^e\2", form)
```

Figure 2: A revised Python implementation of the English e-insertion rule.

Examples from Turkish

inek	'cow'	ineği	'his cow'
kuyruk	'tail'	kuyruğu	'its tail'
köpük	'foam'	köpüğü	'its foam'
yatak	'bed;	yatağı	'its bed'

Table 2: Turkish k/ğ alternation

	'hand'	'köy'	'oda'	'korku'
unmarked	el	köy	oda	korku
accusative	eli	köyü	odayı	korkuyu
genitive	elin	köyün	odanın	korkunun
dative	ele	köye	odaya	korkuya
locative	elde	köyde	odada	korkuda
ablative	elden	köyden	odadan	korkudan

Table 3: Turkish vowel harmony

Examples from Zongozotla Totonac

Consider the forms of the verb *taʃtúh* 'leave' listed in Table 4. The roots are shown in violet. There are three different forms:

- taʃtúh
- taʃtí
- taʃta

The /h/ is only present when the root is at the end of the world. The allomorph with /i/ occurs before prefixes beginning in /y/. The allomorph /taʃta/ occurs before prefixes beginning with /q/. As will become more clear when we talk about phonetics and phonology, this makes a lot of sense. These probably are an example of phonologically conditioned allomorphy. A less-clear case is the past tense prefix */ʃa*. */ʃa* occurs in first person exclusive (1SG and 1PL.EXCL) forms. This could be phonologically conditioned (an /a/ is inserted to prevent a sequence of three consonants in a row from surfacing) or it could be morphologically conditioned (it occurs just in [+me, −you] verbs).

'leave'	Present	Past	Future
1SG	k-taʃtúh	ʃa-k-taʃtúh	na-k-taʃtúh
2SG	taʃtí-ya'	ʃ-taʃtí-ya'	na-taʃtí-ya'
3SG	taʃtúh	ʃ-taʃtúh	na-taʃtúh
1PL.EXCL	k-taʃti-yá'w	ʃa-k-taʃti-yá'w	na-k-taʃti-yá'w
1PL.INCL	taʃti-yá'w	ʃ-taʃti-yá'w	na-taʃti-yá'w
2PL	taʃti-yá:'tit	ʃ-taʃti-yá:'tit	na-taʃti-yá:'tit
3PL	taʃta-qó:y	ʃ-taʃta-qó:y	na-taʃta-qó:y

Table 4: Some person and number inflections of Totonac *taʃtúh* 'leave'.

Other Examples

meng	+	urus	mengurus	'take care'
meng	+	tulis	menulis	'write'
meng	+	kirim	mengirim	'send'
meng	+	pakai	memakai	'use'
meng	+	sewa	menyewa	'rent'

Table 5: Nasal substitution in Indonesian

Suppletive Allomorphy

Consider the verb *go* in English. It's past form is *went*. There is no plausible phonological rule by which these two word forms can be derived from the same basic form. This is called SUPPLETION or SUPPLETIVE ALLOMORPHY. Other examples in English include *good*, *better*, and *best*. These are all forms of the same word, but they derive from different basic forms. This is most clear for *good*, but there is also no general rule by which *better* and *best* can be derived from the same basic form.

Morphologically Conditioned Allomorphy

Sometimes, there are different versions of a morpheme depending on the "class" of the word (like what are called declensions of nouns and conjugations of verbs). Other times, there are different versions of a morpheme depending what other morphemes (and what other inflectional or derivational meanings) occur in a word. Consider the forms of the Totonac verb *paʃ* 'swim' shown in Table 6. The past suffix is *-lh* except in the second person singular, where we have *-t*. To complicate things, in the second person singular, we have *-a'* except in the past, where we have *-t*. It appears that *-t* represents singular and second person and past and that this version of the person morpheme and the tense morpheme surfaces when the word has all three features

'swim'	Present	Perfective	Future
1SG	k-paʃ	k-paʃ-lh	na-k-páʃ
2SG	páʃ-a'	paʃ-t	na-páʃ-a'
3SG	paʃ	paʃ-lh	na-páʃ

Table 6: Some singular forms of Totonac *paʃ* 'swim'

(second person, singular number, and past tense) but the other allomorphs of the two morphemes suface elsewhere.

Implications for Tokenization

All widely used tokenization schemes treat different allomorphs of the same morpheme as different vocabulary items. This is suboptimal, especially for less common morphemes, since embeddings of each of the separate types are likely to be less informative than the embedding of a type that subsumes all of them. Take the case of one of the two negative prefixes in English:

(3) a. **im**balanced
b. **im**patient
c. **im**penetrable
d. **im**ponderable
e. **im**mortal
f. **im**moral

(4) a. **in**ordinate
b. **in**applicable
c. **in**decipherable
d. **in**dissoluble
e. **in**tangible
f. **in**terminable
g. **in**separable
h. **in**secure
i. **in**finite

(5) a. **il**legal
b. **il**liberal
c. **il**logical
d. **il**limitable
e. **il**legible

(6) a. **ir**reversible
b. **ir**revocable
c. **ir**resistible
d. **ir**reproachable
e. **ir**reconcilable
f. **ir**religious
g. **ir**rational
h. **ir**regular

We could learn representations for *im-*, *in-*, *il-*, and *ir-* separately. However, the number of actual examples in our training data will be, in the final analysis, not that large. As a result, the embeddings may, given the vagaries of small numbers, end up being quite different from one another. A tokenization scheme in which allomorphy was factored out would have the advantage of reducing sparsity and increasing generality.

Hypothesis: If, on formal grounds, it can be known that two units, *A* and *B* realize one morpheme *M* (in different context), *A* and *B* should be given the same representation.

Languages are Socially, not Informationally, Optimal

Some languages (e.g., Vietnamese) almost entirely lack allomorphy. There is no obvious sense in which they are less optimal for communication than languages rich in allomorphy (e.g., Totonac). Does allomorphy help language users to communicate information?

Some people claim that phonologically conditioned allomorphy makes words easier to pronounce or sign (or write?) It may do this in some cases, but in other cases, it may actually make language more difficult to articulate. And "ease of articulation" does nothing to explain morphologically-conditioned allomorphy.

(7) Some observations about allomorphy

a. We would never construct a programming language with allomorphy (where, for example, the names of identifiers change in systematic ways depending on context)

b. This is because allomorphy increases the cognitive load placed upon readers and writers of code without (**by definition**) conveying any additional information to the interpreter/compiler

c. The situation for human language users is not different

d. Vietnamese (low allomorphy) can communicate the same set of propositions as Totonac (high allomorphy), but Totonac makes it more difficult

e. But allomorphy does communicate an important kind of information: identity

f. Because allomorphy is difficult to master, it distinguishes children from adults and outsiders from insiders

g. **Allomorphy communicates social information**

h. Vietnamese has other mechanisms for doing this

References