Linguistics 200

Suppose you are working on a phonological analysis, and you discover that there is a pattern, involving predictable information, to be accounted for. You may have found this pattern on the basis of morphological alternations, or complementary distribution, or both.

One way to model predictable information is to propose a phonological rule. A rule needs to use the tools of our phonological model in order to express the following information:

- What natural class of segments is the target, that is, is affected by the rule? (Remember that a single segment is also a natural class, although a very small one.)
- How is that natural class of segments changing? What features are involved?
- Under what circumstances are the segments changing? What are the crucial factors in the segments’ environment that determine where they will or will not change?

Here are some guidelines to follow when you are trying to figure out how best to state a rule in formal rule notation using features.

### 1. Formal rule notation

Formal rule notation works as follows.

(1) Formal Rule Notation: \( A \rightarrow B / X__Y \)

- **A**: The target of the rule, in terms of features
- **B**: The features that are being changed by the rule
- **X__Y**: The environment where the change occurs, expressed using certain conventions:
  - Indicate any crucial segments/natural classes in the environment in terms of features.
  - Use an underscore, "__", to show where the changing segment must be in relation to the crucial segments in the environment. Some examples:
    - after a nasal: \([+\text{nas}]\) __
    - before a lateral: __ \([+\text{lat}]\)
    - between two stops: \([-\text{cont}]\) __ \([-\text{cont}]\)
    - at the beginning of a word: \# __
  - You can show that something in the environment is optional using parentheses, "( )".
  - There is also a special abbreviation used to mean "zero or more, number irrelevant": a subscript zero. We can write \(C_0\) for "zero or more consonants", and \(V_0\) for "zero or more vowels".

The basic mechanics of using formal rule notation are relatively straightforward. However, the more carefully you think about how to use natural classes and features to express meaningful generalizations about the pattern, the more insightful your rule will be.
As a basic rule of thumb, **make your analysis as general as possible** — don’t include any features or natural classes in your rule statement that don’t need to be there to make the rule work. Always remember that one important reason for developing solutions to phonology problems is to explore what characteristics or entities are **crucially necessary** in order to model human mental grammar. Therefore, we want to make our rule statements — and our phonological analyses in general — as simple and elegant as possible, in order to highlight what factors (features, rule types, etc.) really are necessary for capturing the phenomenon under investigation. And if we find that a common, phonetically plausible pattern can only be written as a complex, cumbersome rule, this might indicate a problem with our model itself.

The rest of this handout discusses ways of making rules more general and insightful.

---

### 2. Stating the target of a rule

**A. Stating the target in terms of features**

The **target** of a rule is the natural class of segments that the rule **applies to**. Find it by:

1. **listing out all the segments you want to include in this natural class**
2. **listing out all the segments you do not want to include** — do this based on the inventory of sounds that **appear in the data set** you are working on
3. **putting together a (hopefully small) set of feature values that lets you include all of the segments in (a) while not including any of the segments in (b)**

It is **not a problem** if the rule you write would also apply to additional segments that simply don’t appear in the data set. (In fact, this is a good thing, because it means your rule is general and you are making predictions!) More on this in section 6 below.

**B. Never involve morphology unless it’s absolutely necessary**

Here’s another important point about keeping rules general: **Never assume that a rule will be specific to an individual morpheme** unless there is actual evidence showing that this is the case. (And even if you think there is evidence, first see if you can come up with a different way of analyzing the data set that does not need to refer to particular morphemes. Restricting a phonological rule to specific morphemes should be seen as an absolute last resort that should be avoided whenever possible.)

For example, in **Lamba**, the vowels in the neuter and applied suffixes change in height depending on the height of the preceding vowel.

---

### (2) Initial description of Lamba [i]~[e] alternation

In the neuter and applied suffixes in Lamba, the high front vowel /i/ becomes the mid front vowel [e] when it is preceded by a mid vowel. (Otherwise, it remains unchanged.)

Now, let’s look to see if we can make our analysis any more general.
In fact, we discover that there is no need to restrict our rule to these particular morphemes. There are no vowels anywhere in the data set that would be exceptions to Generalization #1 — that is, no high vowels are ever preceded by mid vowels in the whole data set. So, to make our analysis more general, we revise our description of the phenomenon to apply to the language as a whole, not specifically to the neuter and applied suffixes. (We could test this hypothesis if further investigation into Lamba turned up verb roots with multiple vowels: we are currently predicting that the same generalization will hold within roots too.)

### (3) Morphology-free generalization for Lamba [i]~[e] alternation

In Lamba, high front vowels become mid when they are preceded by a mid vowel.

Another good reason for treating this phenomenon as a general pattern of Lamba is that these two different verb suffixes are seen to behave exactly the same way. If we decided to write morpheme-specific rules, we would need to write one for the neuter suffix and one for the applied suffix, and it would be a coincidence that the two rules did exactly the same thing.

### 3. Stating the change imposed by the rule

Once you have stated the target of the rule as a natural class, in terms of features, the next step is to state the change imposed by the rule — again in terms of features.

A. Only change features that the rule is really changing

When stating the change that a rule carries out, never repeat features that are crucially part of the natural class of the target. For example, if a language changes /s/ to [z] between vowels, you should not write this rule:

\[
\begin{array}{c}
\text{BAD RULE} \\
\text{COR} +\text{cont} \\
-\text{voi} \\
\rightarrow \\
\text{COR} +\text{cont} \\
+\text{voi} \\
/ [-\text{cons}] _{1} [-\text{cons}] \\
\end{array}
\]

(This rule makes /s/ \rightarrow [z] / V_V.)

This rule is bad because it implies that the mental grammar is totally replacing /s/ with some arbitrary sound [z]. An implausible rule changing /s/ to [m] wouldn’t look much different.

What we want to do instead is write a rule that captures our insight that what the mental grammar is really doing in this case is changing the voicing of /s/.

\[
\begin{array}{c}
\text{GOOD RULE} \\
\text{COR} +\text{cont} \\
-\text{voi} \\
\rightarrow \\
[+\text{voi}] \\
/ [-\text{cons}] _{1} [-\text{cons}] \\
\end{array}
\]

(Still achieves /s/ \rightarrow [z] / V_V.)
So we only mention the feature that actually has been changed — [+voi] — and let the outcome of the rule inherit all the other features of the target, unchanged.

B. An apparent exception (but not really): Vacuous rule application

There is one case where it doesn’t hurt to make the change of the rule mention feature values that some of the sounds in the natural class of the target of the rule already have. This is when stating the target of the rule in a maximally general way leads to vacuous (i.e., “empty”) rule application.

This is best illustrated with an example. Suppose, in the imaginary language we were just discussing, there is no phoneme /z/. The only anterior coronal fricative phoneme is /s/. In this case, it obviously won’t hurt to state the target of the rule more generally than we did in (5), since there is no /z/ out there for the rule to apply to. (I.e., remove [–voi] from the target.)

\[
\begin{array}{c|c|c}
\text{COR} & \text{+cont} & \rightarrow \\
\hline
\text{RULE} & & [+voi] / [-cons] [–cons] \\
\end{array}
\]

(Still achieves /s/ \rightarrow [z] / V_V.)

But now suppose a language has /s/ and /z/ as separate phonemes, along with this same rule that turns /s/ into [z] between vowels. (Note that /s/ and /z/ therefore undergo neutralization in this language in the context V_V.) We can still use the rule as stated in (6) even though it will technically apply to /z/ also. Why is this okay?

(a) The rule doesn’t contradict the data set, because “voicing” /z/ still gives [z].

(b) The rule doesn’t technically violate the principle of not repeating unchanged information after the arrow. Why not? Because the feature [+voi] is not a crucial attribute of the natural class that forms the target of this rule. We can tell this because the feature [+voi] isn’t mentioned in the target of the rule — and also because some of the members of the natural class of the rule’s target (as stated very generally in (6)) are voiced and some are not.

In summary, it’s acceptable (and often actually a good sign of a nice, general rule) to include a feature in the statement of the change caused by the rule even if some of the sounds in the target natural class already had that feature value, as long as the feature in question is a necessary outcome of the rule AND wasn’t necessary in specifying the target of the rule.

4. Stating the environment of a rule

Now that you have stated the target and the change, the last step in writing a rule is to state the environment where the rule applies.

As always, it is important to do this as generally as possible. For example, suppose a language turns /m/ into [b] at the beginning of a word, but /m/ appears as [m] every single time it is not at
the beginning of a word. Suppose also that every time /m/ appears at the beginning of a word (and therefore changes to [b]), it also happens to be followed by a vowel.

<table>
<thead>
<tr>
<th>(7) /m/ to [b] in word-initial position</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A. Data set</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>B. Bad rule</strong></td>
</tr>
<tr>
<td><strong>C. Good rule</strong></td>
</tr>
</tbody>
</table>

Why is (7B) a bad rule? Because it isn’t general enough. It happens to be true that the /m/-to-[b] change always happens before a vowel, but you don’t need that information to write a rule that puts [b] in the correct places in this language. All you have to refer to in the environment to write a rule that works is word-initial position — so that is why rule (7C) is better.

Also, the same warning about morphology applies in stating the environment as it does for stating the target of a rule: It should be something you do only as an absolute last resort, when you have tried all other possibilities and proven that they do not work.

5. Checking a rule for counterexamples

How do we check to make sure our rule works? We need to apply it, perfectly literally (you could even say mindlessly), to the data set:

(a) Find every sound in the data set whose UR (or intermediate form, if a prior rule has already applied) matches the natural class of the rule’s target.

(b) Check to see if that sound stands in the rule’s environment.

(c) If so, check to make sure that the rule has applied to the sound in question.

If you find a sound to which the rule as written will apply, but is not supposed to according to the actual data set, then there is something wrong with your analysis. Maybe the rule isn’t stated quite specifically enough. Or, maybe you have made the wrong hypothesis about the UR of some morpheme or some natural class of sounds. Checking your rule for counterexamples is an important step in any phonological analysis, because it helps you find problems like these.

6. More about generalizing rules

A. Combining individual rules to make a more general analysis

Now that you have stated your rule, and it works, it is time to think about whether this rule stands alone, or whether it should be made part of a more general rule along with some of the other rules in your analysis.
Two (or more) rules can be combined into one more general rule if:

(a) The **changes** imposed by the rules are the **same**.
(b) The **environments** where the rules apply are the **same**.
(c) The **targets** of each rule can be **combined into one more general natural class** that doesn’t incorrectly apply to other sounds in the data set.

If you have two rules that look like they are very similar, but they don’t quite match up, try to see if any of the natural classes (target, change, environment) you have stated for any of your rules could be made slightly more (or, if necessary, slightly less) general so that the different rules begin to resemble each other more closely.

B. Predictions about sounds that are not in the data set

If you follow the preferred strategy and state all your natural classes in terms that are as general as is consistent with the data, you may find that sometimes you are making claims about segments that do not appear in the data. This is a good thing — what you are in fact doing is forming a hypothesis about how such sounds would behave if they were present.

The only time you need to make a natural class more specific is if there are actual **counterexamples** to the more general statement in the data. For example, in the Arabic **consonants** problem, describing the segments in group (d) as “coronal stops” would be too general, because [n] is also a coronal stop but it belongs to group (e). That is, [n] is a counterexample to the proposal that the relevant natural class is simply defined by [–cont, cor]. This justifies referring to the more specific natural class [–son, –cont, cor], coronal **obstruent** stops (alternatively, referring to the property “oral” ([–nas]) would work here too).

Here is an example of a rule that can be made more general, and therefore makes predictions about sounds that are not seen (in the relevant context) in the data set. Going back to the [i]~[e] alternation in Lamba, we note that the crucial factor in the “morphology-free generalization” in (3) seems to be **vowel height**. Now, we might wonder if there is any reason to restrict our statement to apply only to high front vowels. In fact, the answer is no. We have no information about high back vowels occurring after mid vowels, so there is no counterevidence against making our rule apply to all high vowels (front and back).

So, let’s restate our description in yet more general terms. We are now making a hypothesis about how back vowels behave — a hypothesis that can be tested against further data.

**Fully general statement of the Lamba [i]~[e] alternation**

In Lamba, high vowels become mid when they are preceded by a mid vowel.

Do we even need to restrict our statement to high vowels, or can we say that all vowels become mid when a mid vowel precedes? This time, we see that we can’t go that far. There are examples in the data set that show that a low vowel remains unchanged even when it follows a
mid vowel. So, we seem to have arrived at the final version of our descriptive statement of the pattern.

Here is how we can state our rule for vowels in Lamba using formal rule notation.

(9) Rule for Lamba vowels

\[
[+hi] \rightarrow [-hi] / [-hi] _{-lo} C_0 __
\]

C. Choosing between equally general ways of stating a rule

Sometimes there are two (or more) ways of stating a rule that are (approximately) equally general and simple. In such a case, see if one or the other version of the rule seems more insightful. In particular, does it express something about the phonetic plausibility of the rule — does it indicate something about what is special about that particular feature change given that particular target or that particular environment?

For example, the rule for Lamba vowels in (9) above can actually be written another way, using a different feature specification for the target of the rule: \([-lo]\) instead of \([+hi]\).

(10) Rule for Lamba vowels, second option

\[
[-lo] \rightarrow [-hi] / [-hi] _{-lo} C_0 __
\]

This second version of the rule will of course apply to mid vowels as well as high vowels, but this is a case of vacuous rule application (mid vowels are already \([-hi]\)), so it’s not a problem.

Either way of stating this rule can be considered correct; both are equally general in terms of how many features we must use to designate the relevant natural classes. However, there might be a reason to prefer (10). This statement of the rule emphasizes that the rule wants a \([-lo]\) vowel to change in a particular way, if a certain kind of \([-lo]\) vowel precedes it. That is, it brings out a connection between the target and the environment of the rule, in a way that isn’t as clearly seen in the version of the rule in (9). This version of the rule also highlights the fact that the rule’s change, to \([-hi]\), is exactly the change needed to bring the target \([-lo]\) sound into conformity with the \([-hi, -lo]\) sound that is the crucial factor in the environment.