Overview

• How lexical rules fit in
• Three types of lexical rules, constraints
• Example: Plural noun lexical rule
• Advice on writing lexical rules
• Constant lexemes
• ARG-ST & ARP
• The feature FORM
Lexical Types & Lexical Rules

• Lexemes capture the similarities among run, runs, running, and ran

• The lexical type hierarchy captures the similarities among run, sleep, and laugh, among those and other verbs like devour and hand, and among those and other words like book.

• Lexical rules capture the similarities among runs, sleeps, devours, hands, ...
Parsimony & Plausibility

• Lexical rules capture **productive** generalizations.

• There may be some ‘precompiling’ going on as well.
Three Kinds of Lexical Rules

- Inflectional: *lexeme* to *word*
  Examples?

- Derivational: *lexeme* to *lexeme*
  Examples?

- Post-Inflectional: *word* to *word*
  (Chapters 11, 13, 14)
Three Subtypes of \( l \)-rule

\[ l \text{-rule} \]

\[ i \text{-rule} \quad d \text{-rule} \quad p_i \text{-rule} \]

\[ l \text{-rule} : \begin{cases} \text{INPUT} & \text{l-sequence} \langle X, [\text{SEM} / 2] \rangle \\ \text{OUTPUT} & \text{l-sequence} \langle Y, [\text{SEM} / 2] \rangle \end{cases} \]

\[ i \text{-rule} : \begin{cases} \text{INPUT} & \langle X, \begin{bmatrix} \text{lexeme} \text{ SYN} & 3 \\ \text{ARG-ST} & A \end{bmatrix} \rangle \\ \text{OUTPUT} & \langle Y, \begin{bmatrix} \text{word} \text{ SYN} & 3 \\ \text{ARG-ST} & A \end{bmatrix} \rangle \end{cases} \]

\[ d \text{-rule} : \begin{cases} \text{INPUT} & \langle X, \begin{bmatrix} \text{lexeme} \text{ SYN} & \end{bmatrix} \rangle \\ \text{OUTPUT} & \langle Y, \begin{bmatrix} \text{lexeme} \text{ SYN} & \end{bmatrix} \rangle \end{cases} \]
Plural Noun LR

\[
\begin{align*}
\text{i-rule} & \\
\text{INPUT} & \langle \underline{1}, \text{cntn-lxm} \rangle \\
\text{OUTPUT} & \langle F_{NPL}(\underline{1}) , \begin{bmatrix} \text{word} \\
\text{SYN} \begin{bmatrix} \text{HEAD} \begin{bmatrix} \text{AGR} \begin{bmatrix} \text{NUM pl} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \rangle
\end{align*}
\]
Plural Noun LR with Inherited Constraints

\[ \text{i-rule} \]

INPUT \[ \langle 1 \rangle, \]
\[ \begin{array}{c}
\text{SYN} \quad 3 \\
\text{VAL} \\
\text{SEM} \quad 2 \text{[MODE / ref]} \\
\text{ARG-ST} \quad B \oplus C
\end{array} \]

\[ \text{cntn-lxm} \]
\[ \begin{array}{c}
\text{HEAD} \\
\text{AGR} \quad 4 \text{[PER 3rd]} \\
\text{SPR} \quad \langle \text{DP} \quad \langle \text{COUNT +} \rangle \rangle \\
\text{ARG-ST} \quad B \oplus C
\end{array} \]

\[ \text{noun} \]
\[ \begin{array}{c}
\text{HEAD} \quad \langle \text{AGR} \quad \langle \text{NUM pl} \rangle \rangle \\
\text{VAL} \\
\text{SPR} \quad \langle \text{COMPS} \quad [B, C] \rangle \\
\text{ARG-ST} \quad B \oplus C
\end{array} \]

\[ \langle 1 \rangle \]

\[ \text{F}_{NPL}(\Pi) \]
\[ \begin{array}{c}
\text{SYN} \quad 3 \\
\text{VAL} \\
\text{SEM} \quad 2 \\
\text{ARG-ST} \quad B \oplus C
\end{array} \]

\[ \langle 1 \rangle \]
Practicalities - Applying Lexical Rules

- INPUT is a family of lexical sequences.
- OUTPUT is another family of lexical sequences.
  - ...usually a smaller family
  - ...usually a disjoint one
- The only differences between the families are those stipulated in the rule (or the rule’s type).
- Similarities are handled by the constraints on $l$-rule and its subtypes.
- If we’ve written the LRs correctly, nothing is left underconstrained.
Example: Lexical Entry for *cat*

\[
\langle \text{cat} , \left[ \begin{array}{c}
\text{cntn-lxm} \\
\text{SEM} \\
\text{RESTR} \\
\end{array} \right] \langle [\text{RELN} \ \text{cat}] \rangle \rangle
\]
Example: *cat*, with inheritance

\[
\begin{align*}
\langle \text{cat} , \rangle \\
\langle \text{cntn-lxm} \rangle \\
\langle \text{SEM} \rangle \\
\langle \text{ARG-ST} \rangle \\
\langle \text{cat} \rangle \\
\langle \text{noun} \rangle \\
\langle \text{head} \rangle \\
\langle \text{val} \rangle \\
\langle \text{spr} \rangle \\
\langle \text{spr} \rangle \\
\langle \text{sem} \rangle \\
\langle \text{syn} \rangle \\
\langle \text{cntn-lxm} \rangle
\end{align*}
\]
Plural Noun LR

\[i-rule\]

INPUT \[\langle 1, cntn-lxm \rangle \]

OUTPUT \[\langle F_{NPL}(1), \left[ \begin{array}{c} \text{word} \\ \text{ SYN } \text{ HEAD } [ \text{ AGR } [ \text{ NUM pl } ] ] \end{array} \right] \rangle \]
Licensing \textit{cats}

\begin{itemize}
\item \textbf{i-rule}
\item \textbf{cntn-lxm}
\item \textbf{INPUT} \langle [1] \text{cat}, \rangle
\item \textbf{SEM} \langle [1] \text{cat}, \rangle
\item \textbf{ARG-ST} \langle X \rangle \oplus \langle C \rangle
\item \textbf{OUTPUT} \langle \text{F}_{NPL}(1), \rangle
\item \textbf{SEM} \langle [2] \rangle
\item \textbf{ARG-ST} \langle [1] \rangle
\end{itemize}
Three Subtypes of \( l \)-rule

\[ l \text{-rule} \]

\[ i \text{-rule} \quad d \text{-rule} \quad pi \text{-rule} \]

\[ l \text{-rule} : \begin{bmatrix}
\text{INPUT} & l \text{-sequence} \langle X, [\text{SEM} / 2] \rangle \\
\text{OUTPUT} & l \text{-sequence} \langle Y, [\text{SEM} / 2] \rangle
\end{bmatrix} \]

\[ i \text{-rule} : \begin{bmatrix}
\text{INPUT} & \langle X, \begin{bmatrix}
\text{lexeme SYN} \quad 3 \\
\text{ARG-ST} \quad A
\end{bmatrix} \rangle \\
\text{OUTPUT} & \langle Y, \begin{bmatrix}
\text{word SYN} \quad 3 \\
\text{ARG-ST} \quad A
\end{bmatrix} \rangle
\end{bmatrix} \]

\[ d \text{-rule} : \begin{bmatrix}
\text{INPUT} & \langle X, \begin{bmatrix}
\text{lexeme SYN} \quad 3
\end{bmatrix} \rangle \\
\text{OUTPUT} & \langle Y, \begin{bmatrix}
\text{lexeme SYN} \quad 3
\end{bmatrix} \rangle
\end{bmatrix} \]
**cats**: The Lexical Sequence

\[
\langle \text{cats} , \rangle
\]

\[
\begin{array}{l}
\langle \text{cats} , \rangle \\
\text{word} \\
\text{SYN} \\
\text{VAL} \\
\text{SEM} \\
\text{ARG-ST}
\end{array}
\]

\[
\begin{array}{l}
\text{word} \\
\text{HEAD} \\
\text{SPR} \\
\text{MODE} \\
\text{INDEX} \\
\text{RESTR}
\end{array}
\]

\[
\begin{array}{l}
noun \\
\text{AGR} 3pl \\
\text{COUNT} + 7 \\
\text{cat} \\
\text{k}
\end{array}
\]

© 2003 CSLI Publications
Practicalities -- Writing Lexical Rules

• Determine the type of the LR.
• Determine the class of possible inputs.
• Determine what should change.
  • If INPUT and OUTPUT values are identified (by default or otherwise) and only OUTPUT value is mentioned, then... information is added.
    (Lexical sequences incompatible with that value are not possible inputs)
  • If INPUT and OUTPUT values are identified by default, but different values are given on the INPUT and OUTPUT of the rule, then... information is changed.
  • If INPUT and OUTPUT values are identified by an inviolable constraint, but different values are given on the INPUT and OUTPUT of the rule, then... there is no well-formed output
Constant lexemes

• What kinds of words are constant lexemes in our grammar?

• Why do we need a rule for these words?

• What would be an alternative analysis?
What keeps this from applying to, say, verb lexemes?

Why is this an $i$-rule?
ARG-ST & ARP

• Given the ARP, what do we need to specify about the valence properties of words?

• Why isn’t the ARP a constraint on the type *lexeme*?
The Feature FORM

• Different inflected forms of verbs show up in different syntactic environments. Examples?

• These different forms are syntactically distinguished by the feature FORM, as assigned by lexical rules.

• FORM is also useful in our analyses of coordination and PP selection.
What rules these out?

- *Kim eat pizza.
- *Kim seems to eats pizza.
- *Dana helped Leslie pack and moved.
- *Kim relies for Sandy.
- *Dana walked and Kim.
Overview

• How lexical rules fit in
• Three types of lexical rules, constraints
• Example: Plural noun lexical rule
• Advice on writing lexical rules
• Constant lexemes
• ARG-ST & ARP
• The feature FORM