Computational Morphology

Morphology

Study of words, how they are formed from smaller meaningful units called **morphemes**, and their relationship to other words in the same language.

Pizzas

- 2 morphemes, 'pizza' and 's'
- 's' is a plural marker on nouns

Untouchables

Four morphemes

- un- 'not'
- touch 'an act of touching someone or something'
- -able 'capable of'
- 's' is a plural marker on nouns

Type of Morphemes

Allomorphs

Variants of the same morpheme, but cannot be replaced by one another

- opposite: un-happy, in-comprehensible, im-possible, ir-rational

Bound Morphemes

Cannot appear as a word by itself.

-s (dog-s), -ly (quick-ly), -ed (walk-ed)

Free Morphemes

Can appear as a word by itself; often can combine with other morphemes too.

house (house-s), walk (walk-ed), of, the, or

Type of Morphemes

Stems and Affixes

- Stems: The core meaning bearing units
- Affixes: Bits and pieces adhering to stems to change their meanings and grammatical functions

Mostly, stems are free morphemes and affixes are bound morphemes

Types of affixes

- Prefix: un-, anti-, etc (a-, ati-, pra- etc.)
  un-happy, pre-existing
- Suffix: -ity, -ation, etc (-taa, -ke, -ka etc.)
  talk-ing, quick-ly
- Infix: '*n*' in '*vindati*' (he knows), as contrasted with *vid* (to know).
  Philippines: basa 'read' → b-um-asa 'read'
  English: abso-bloody-lutely (emphasis)
- Circumfixes - precedes and follows the stem
  Dutch: berg 'mountain', ge-berg-te 'mountains'

Type of Morphemes

Content morphemes

Carry some semantic content

car, -able, un-

Type of Morphemes

Content morphemes

Carry some semantic content

car, -able, un-

Functional morphemes

Provide grammatical information

-s (plural), -s (3rd singular)

The two relationships

Two different kind of relationship among words

Inflectional morphology

Grammatical: number, tense, case[a], gender

Creates new forms of the same word: *bring, brought, brings, bringing*

[a] https://www.quora.com/What-is-the-difference-between-he-and-him

Derivational morphology

Creates new words by changing part-of-speech: *logic, logical, illogical, logically, logician, logicize*

The two relationships

Two different kind of relationship among words

Inflectional morphology

Grammatical: number, tense, case[a], gender

Creates new forms of the same word: *bring, brought, brings, bringing*

[a] https://www.quora.com/What-is-the-difference-between-he-and-him

Derivational morphology

Creates new words by changing part-of-speech: *logic, logical, illogical, logically, logician, logicize*

Fairly systematic but some derivations missing: *sincere - sincerity, scarce - scarcity, curious - curiosity, fierce - fiercity?*

Morphological processes

Concatenation

Adding continuous affixes - the most common process:

- hope+less, un+happy, anti+capital+ist+s

Often, there are phonological/graphemic changes on morpheme boundaries:

- book + s [s], shoe + s [z]
- happy +er → happier

Suppletion

'irregular' relation between the words

go - went, good - better

Morpheme internal changes

The word changes internally

sing - sang - sung, man - men, goose - geese

Morphological processes

Reduplication: part of the word or the entire word is doubled

- Nama: ‘go’ (look), ‘go-go’ (examine with attention)
- Tagalog: ‘basa’ (read), ‘ba-basa’(will read)
- Sanskrit: ‘pac’ (cook), ‘papa¯ca’ (perfect form, cooked)
- Phrasal reduplication (Telugu): *pillava¯d.u nad.ustu¯ nad.ustu¯ pad.i po¯ya¯d.u*
  (The child fell down while walking)

Morphological processes

Reduplication: part of the word or the entire word is doubled

- Nama: ‘go’ (look), ‘go-go’ (examine with attention)
- Tagalog: ‘basa’ (read), ‘ba-basa’(will read)
- Sanskrit: ‘pac’ (cook), ‘papa¯ca’ (perfect form, cooked)
- Phrasal reduplication (Telugu): *pillava¯d.u nad.ustu¯ nad.ustu¯ pad.i po¯ya¯d.u* (The child fell down while walking)

Any examples in Hindi?

Morphological processes

Reduplication: part of the word or the entire word is doubled

- Nama: ‘go’ (look), ‘go-go’ (examine with attention)
- Tagalog: ‘basa’ (read), ‘ba-basa’(will read)
- Sanskrit: ‘pac’ (cook), ‘papa¯ca’ (perfect form, cooked)
- Phrasal reduplication (Telugu): *pillava¯d.u nad.ustu¯ nad.ustu¯ pad.i po¯ya¯d.u* (The child fell down while walking)

Any examples in Hindi?

- Baithe baithe
- Khana vana
- Parathe sarathe
- ...

https://www.academia.edu/19403461/Echo-word_formation_in_Hindi

Word Formation

Compounding

Words formed by combining two or more words

Example in English:

- Adj + Adj → Adj: bitter-sweet
- N + N → N: rain-bow
- V + N → V: pick-pocket
- P + V → V: over-do

Particular to languages

room-temperature: Hindi translation?

Word Formation

Acronyms

laser: Light Amplification by Simulated Emission of Radiation

Blending

Parts of two different words are combined

- breakfast + lunch → brunch
- smoke + fog → smog
- motor + hotel → motel

Clipping

Longer words are shortened

doctor, laboratory, advertisement, dormitory, examination, bicycle, refrigerator

Compounding in Social Media

Compounding in Social Media

Mainly in **Twitter hashtags**

- #AmazonPrimeDay
- #MondayMotivation
- #BigBillionDay
- #YouKnowItsRealWhen
- #TheBestFeelingInARelationship
- #RelationshipTips

Processing morphology

- Lemmatization: word → lemma
  saw → {see, saw}
- Morphological analysis : word → setOf(lemma +tag)
  saw → { <see, verb.past>, <saw, noun.sg>}
- Tagging: word → tag, considers context
  Peter *saw* her → { <see, verb.past>}
- Morpheme segmentation: de-nation-al-iz-ation
- Generation: see + verb.past → saw

What are the applications?

- Text-to-speech synthesis:
  lead: verb or noun?
  read: present or past?
- Search and information retrieval
- Machine translation, grammar correction

Morphological Analysis

Input	Morphological Parsed Output
cats	cat +N +PL
cat	cat +N +SG
cities	city +N +PL
geese	goose +N +PL
goose	(goose +N +SG) or (goose +V)
gooses	goose +V +3SG
merging	merge +V +PRES-PART
caught	(catch +V +PAST-PART) or (catch +V +PAST)

Goal

To take input forms like those in the first column and produce output forms like those in the second column.

Output contains stem and additional information; +N for noun, +SG for singular, +PL for plural, +V for verb etc.

Issues involved

boy → boys

fly → flys → flies (y → i rule)

Toiling → toil

Duckling → duckl?

- Getter → get + er
- Doer → do + er
- Beer → be + er?

Knowledge Required

Knowledge of stems or roots

Duck is a possible root, not *duckl*.

We need a dictionary (lexicon)

Morphotactics

Which class of morphemes follow other classes of morphemes inside the word?

Ex: plural morpheme follows the noun

Only some endings go on some words

- *Do+er*: ok
- *Be+er*: not so

Spelling change rules

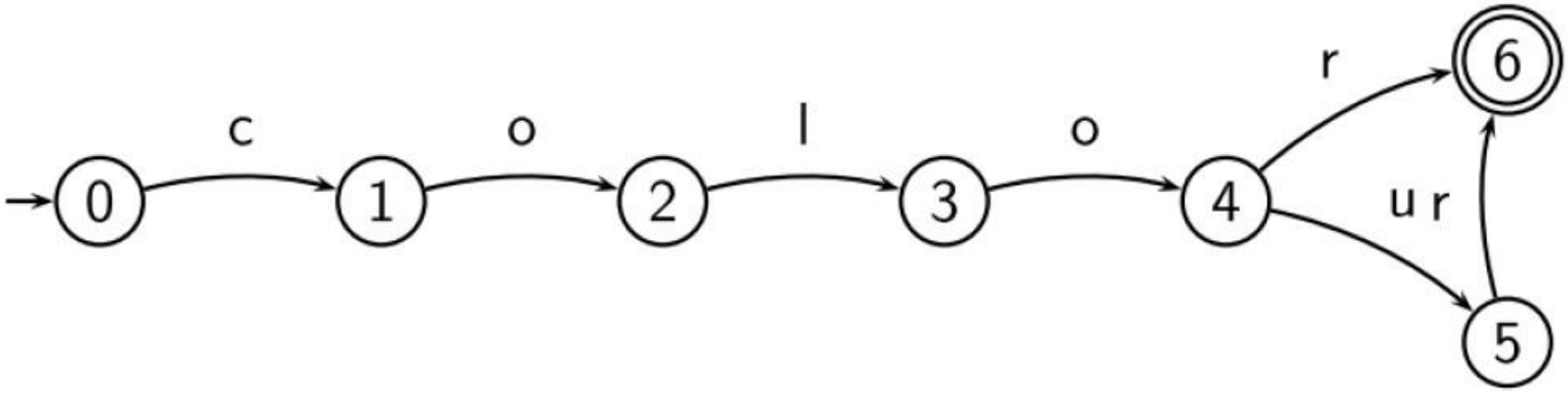
- Get + er → getter
- Fox + s → foxes

Why can't this be put in a big lexicon?

- English: just 317,477 forms from 90,196 lexical entries, a ratio of 3.5:1
- Sanskrit: 11 million forms from a lexicon of 170,000 entries, a ratio of 64.7:1
- New forms can be created, compounding etc.

One of the most common methods is finite-state-machines

Finite State Automaton (FSA)



What is FSA?

- A kind of directed graph
- Nodes are called states, edges are labeled with symbols (possibly empty Z)
- Start state and accepting states
- Recognizes regular languages, i.e., languages specified by regular expressions

FSA for nominal inflection in English

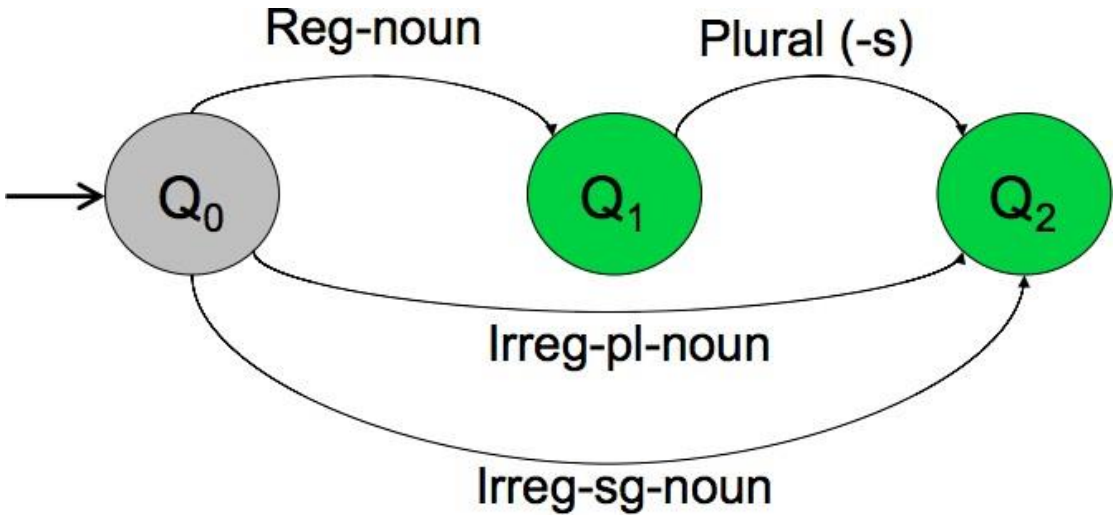
Regular noun examples

- Cat: cats
- Boy: boys
- Bag: Bags

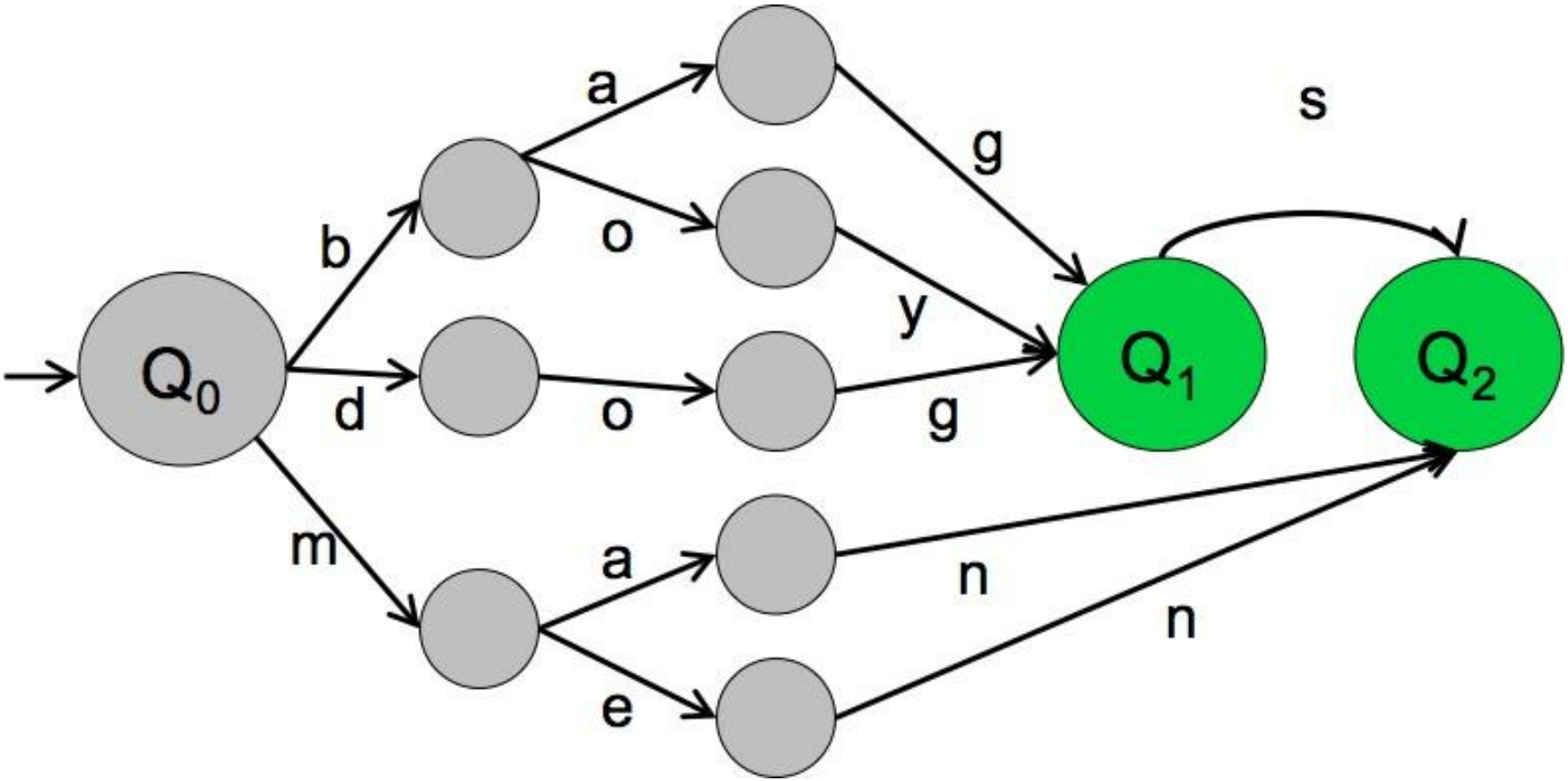
Irregular noun examples

- Calf: calves
- Knife: knives
- Child: children
- Goose: geese

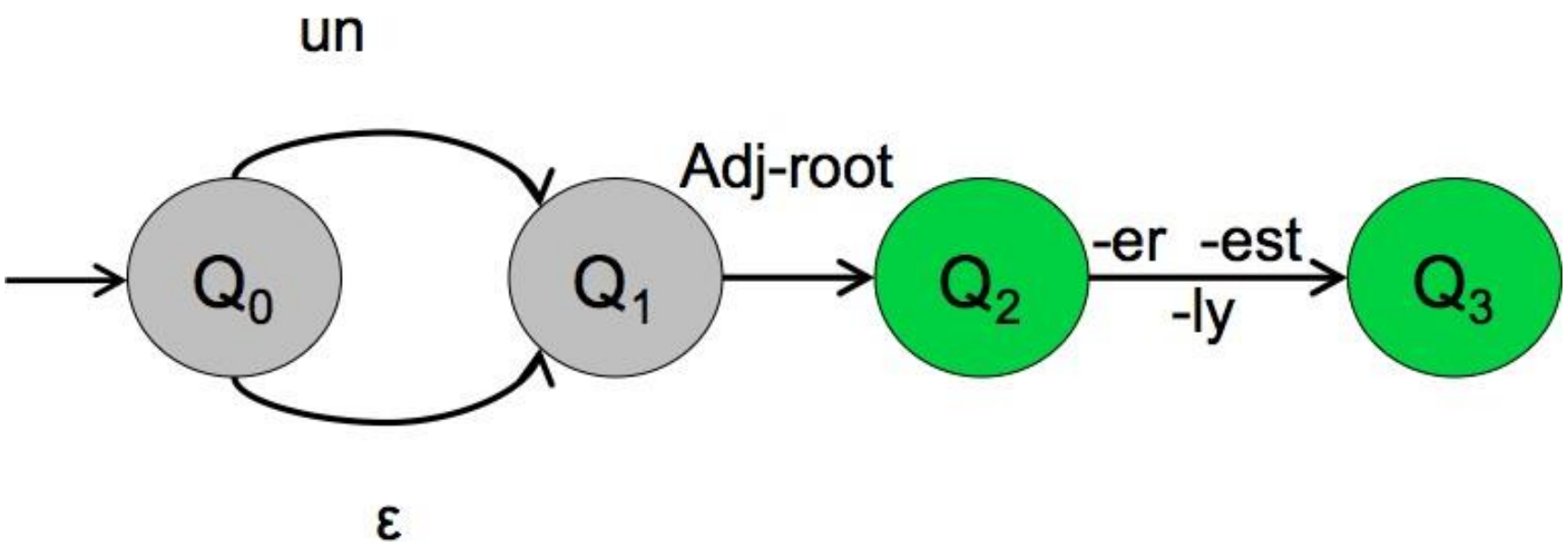
FSA for nominal inflection in English



FSA for nominal inflection in English



FSA for English Adjectives



Word modeled

happy, happier, happiest, real, unreal, cool, coolly, clear, clearly, unclear, unclearly, ...

Caveats

Advantages

- Recognizing problem can be solved in linear time (independent of the size of the automaton)
- There is an algorithm to transform each automaton into a unique equivalent automaton with the least number of states.

Challenges

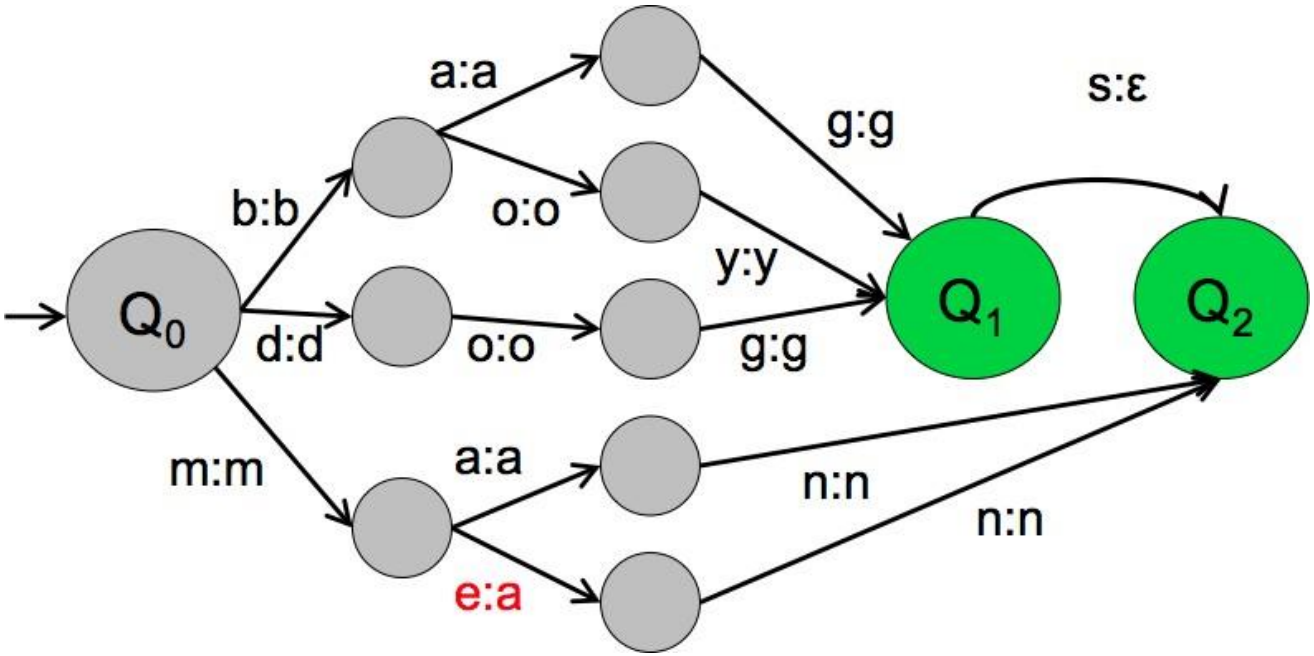
- FSAs are language recognizers/generators.
- We cannot perform Morphological Analysis

Finite State Transducers

- Translate strings from one language to strings in another language
- Like FSA, but each edge is associated with two strings

Finite State Transducers

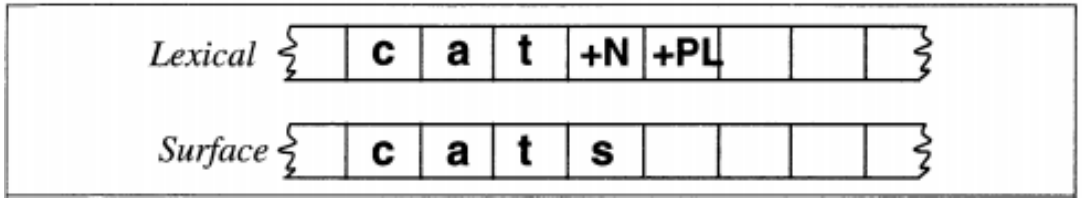
- Translate strings from one language to strings in another language
- Like FSA, but each edge is associated with two strings



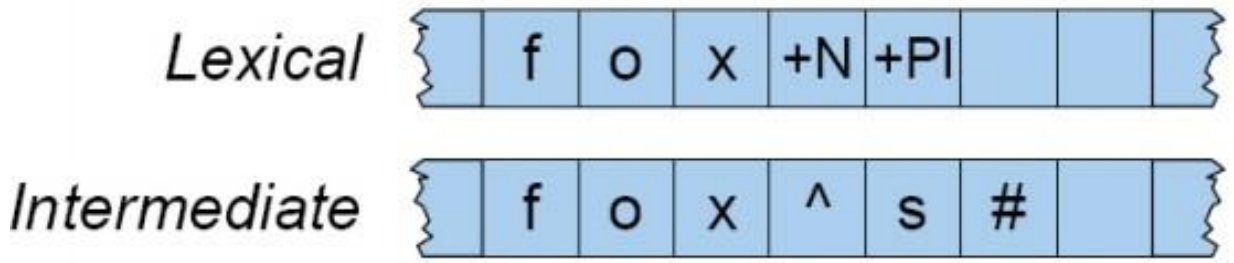
Two-level morphology

Given the input *cats*, we would like to output *cat+N+PL*, talling us that cat is a plural noun.

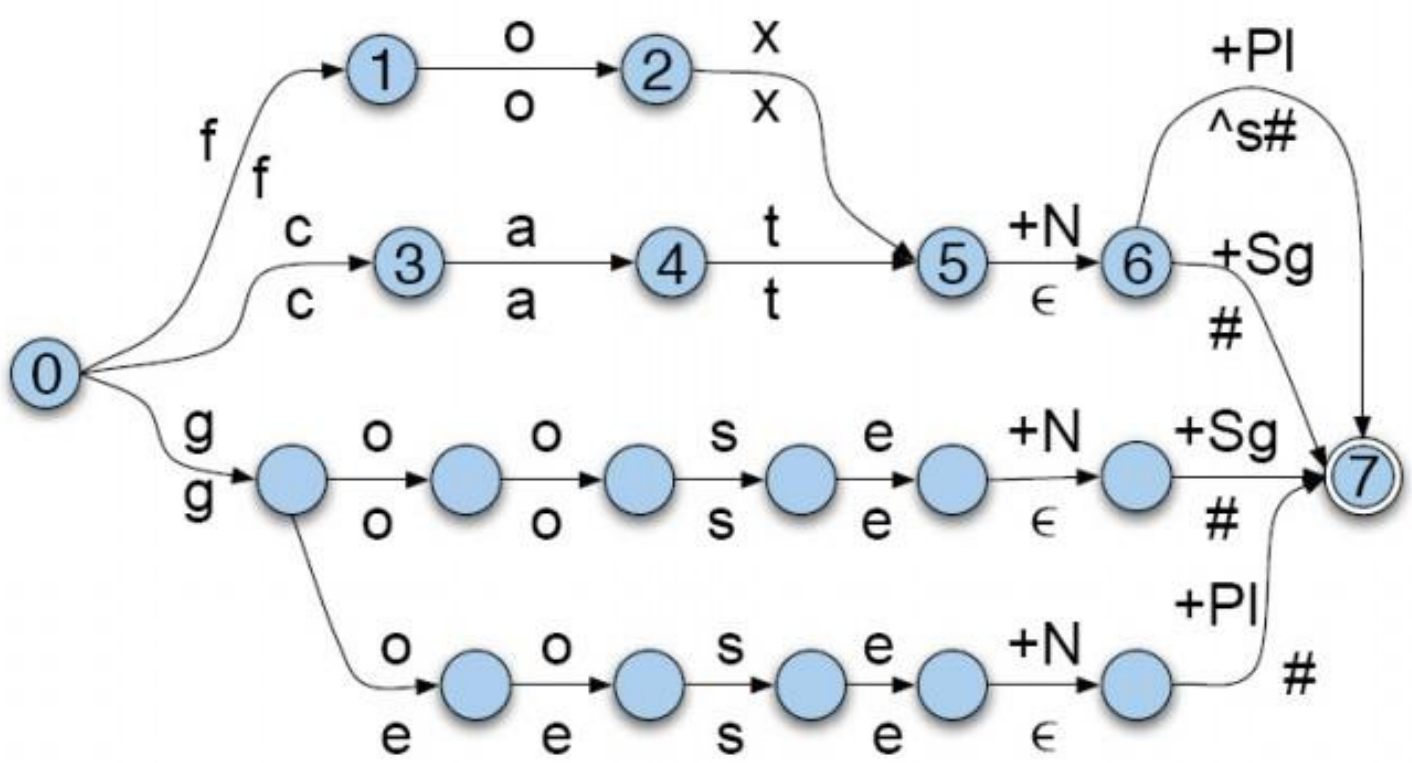
We do this via a version of **two-level morphology**, a correspondence between a lexical level (morphemes and features) to a surface level (actual spelling).



Intermediate tape for Spelling change rules

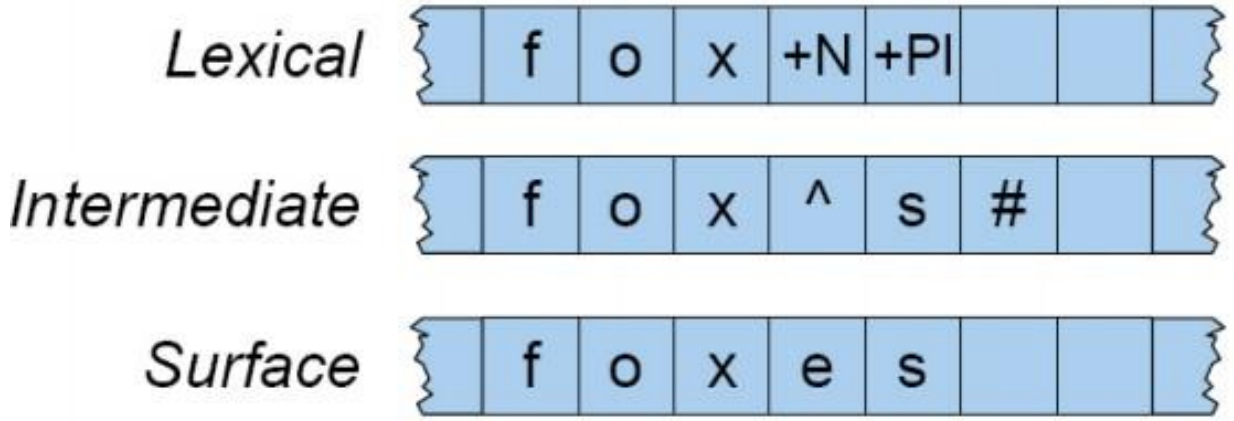


English Nominal Inflection FST



Spelling Handling

A spelling change rule would insert an e only in the appropriate environment.



Rule Handling

Rule Notation

$a \rightarrow b / c\_d$: "rewrite a as b when it occurs between c and d."