MENOUFIA UNIVERSITY
FACULTY OF COMPUTERS AND INFORMATION

Fourth Year (Second Semester)
CS Dept., (CS 436)

Natural Language Processing NLP

Lecture Four

Dr. Hamdy M. Mousa

MORPHOLOGY AND FINITE-STATE TRANSDUCERS

Singulars & Plurals

- Hunting for the plurals of these animals takes more than just tacking on an ***s***.
- *The plural of*
  - *Fox is foxes*
  - *Goose is geese.*
  - *Fish don't usually* change their form when they are plural *one fish, two fish, red fish).*

Singulars & Plurals

- Why don't we just list all the **plural** forms of English nouns, and all the -**ing** forms of English verbs in the dictionary?

- **Singulars and plurals:**
  - **Spelling rules tell =>English words ending in *–y*** are pluralized by changing the ***-y to -i- and adding an -es.***
  - **Morphological rules tell =>*fish has a null plural, and that the plural of goose is formed*** by changing the vowel.

Singulars & Plurals

- The ***problem of recognizing*** that *foxes breaks down into the two morphemes fox and -es is called* ***morphological parsing*****.**
- The plural form of ***these new nouns*** depends on the spelling/pronunciation of the singular form.
- ***Example:*** if the noun ends in -z then the plural form is -es rather than -s.

Morphology

- **Morphology:** the study of meaningful parts of words and how they are put together.
- **Morpheme** is often defined as the minimal meaning-bearing unit in a language.
  - **Morphemes:** are the smallest meaningful spoken units of language.

Example: fox → **single** morpheme (fox)
cats → **two** morphemes (cat and –s)

Example:

- books: two **morphemes** (book and s) but one **syllable**.
- Unladylike: three morphemes, four syllables.

Syllablization

- **Syllable:** a unit of pronunciation having one vowel sound.

- How many syllables are in a word and how do you divide that word into syllables?
  - bin (1 syllables),
  - number (2 syllables),
  - credentials (3 syllables),
  - unkindly (3 syllables),
  - designation (4 Syllables),
  - recognition (4 Syllables)
  - classification (5 syllables),
  - identification (6 syllables)
  - identifications (6 syllables)
  - hypersensitivity (7 syllables),

Morphology

- distinguish two broad classes of morphemes: **stems** and **affixes**.
- Morphological parsing is the task of recognizing the morphemes inside a word.
  - e.g., *hands, foxes, children*
- **Important for many tasks**
  - machine translation
  - information retrieval
  - lexicography
  - any further processing (e.g., part-of-speech tagging)

Root

- **Root**
  - The portion of the word that:
    - is common to a set of derived or inflection forms,
      - if any, when all affixes are removed
    - is not further **analyzable** into meaningful elements
    - carries the principle portion of meaning of the words.

الجذر: هو الوحدة المعجمة الأولية لكل كلمة، التي تحمل المعنى الأهم من تلك الكلمة والدلالة، والتي لا يجوز أن تتجزأ .

Basic Terminologies

- Wordform
  - full inflected or derived form of a word
  - Amusing, amused, amusement etc
- **Lemma**
  - Dictionary form (canonical form) of each word
    - amusements => amusement
    - amused = > amuse
- **Stem**
  - the part of the word that never changes even when morphologically inflected
  - amused, amusing, amusement => amus

Stem

- **Stem**
  - The form of a word after all affixes are removed
  - The root or roots of a word, together with any derivational affixes, to which inflectional affixes are added.
  - In one usage a stem is a form to which affixes can be attached.
  - In the English word ***friendships*** contains the stem *friend*, to which the derivational suffix -*ship* is attached to form a new stem *friendship*, to which the inflectional suffix -*s* is attached.

Affix

- **Affix**
  - A bound morpheme that is joined before, after, or within a root or stem.
  - **Affixes:**
    - Prefixes precede the stem,
    - suffixes follow the stem,
    - circumfixes do both,
    - infixes are inserted inside the stem.
- **Clitic**
  - a morpheme that functions syntactically like a word, but does not appear as an independent phonological word
    - English: ***I've*** *(the morpheme 've is a clitic)*

Word

- **Word Classes**
  - **Parts of speech:** noun, verb, adjectives, etc.
  - Word class dictates how a word combines with morphemes to form new words
- Finite-state methods are particularly useful in dealing with a lexicon
- Many devices need access to large lists of words
  - Spell checkers
  - Syntactic parsers
  - Language Generators
  - Machine translation systems

Inflection & Derivation

Combine morphemes to create words:

- **Inflection:** combination of a word stem with a grammatical morpheme same word class,

  e.g. clean (verb), clean-ing (verb)
  - **Doesn't change the word class**
  - **Usually produces a predictable meaning.**
- **Derivation:** combination of a word stem with a grammatical morpheme, Yields different word class, e.g. clean (verb), clean-ing (noun)
- **Compounding:** combination of multiple word stems.

Inflectional Morphology

- **Inflectional Morphology**

  word stem + grammatical morpheme e.g. cat + s

  - **only for nouns, verbs, and some adjectives**
- **Nouns**
  - **plural:**

    regular: +s, +es irregular: mouse - mice; ox - oxen
  - **rules for exceptions: e.g. -y → -ies like: butterfly - butterflies**
- **possessive: +'s, +'**
- **Verbs**
  - **main verbs (sleep, eat, walk)**
  - **modal verbs (can, will, should)**
  - **primary verbs (be, have, do)**

Inflectional Morphology

Verb Inflections for:

main verbs (sleep, eat, walk); primary verbs (be, have, do)

Morpholog. Form	Regularly Inflected Form			
stem	walk	merge	try	map
-s form	walks	merges	tries	maps
-ing participle	walking	merging	trying	mapping
past; -ed participle	walked	merged	tried	mapped

Morph. Form	Irregularly Inflected Form		
stem	eat	catch	cut
-s form	eats	catches	cuts
-ing participle	eating	catching	cutting
-ed past	ate	caught	cut
-ed participle	eaten	caught	cut

Inflectional Morphology

Noun Inflections for:

regular nouns (cat, hand); irregular nouns(child, ox)

Morpholog. Form	Regularly Inflected Form	
stem	cat	hand
plural form	cats	hands

Morph. Form	Irregularly Inflected Form	
stem	child	ox
plural form	children	oxen

Derivational Morphology

- Derivation in English is quite complex.
- Derivation is the combination of a word stem with a grammatical morpheme, usually resulting in a word of a different class, often with a meaning hard to predict exactly.

Derivational Morphology

- Nominalization is very common kind of derivation in English is **the formation of new nouns**, often from verbs or adjectives.
- Nominalization (formation of nouns from other parts of speech, verbs):

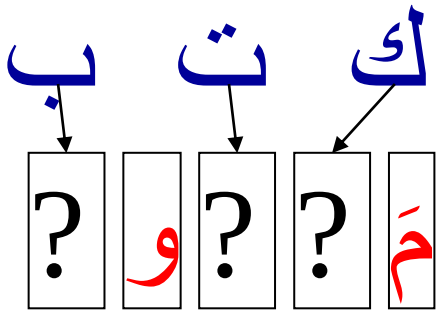
Suffix	Base Verb/Adjective	Derived Noun
-ation	computerize (V)	computerization
-ee	appoint (V)	appointee
-er	kill (V)	killer
-ness	fuzzy (A)	fuzziness

Derivational Morphology

- Nominalization is very common kind of derivation in English is **the formation of new nouns**, often from verbs or adjectives.
- Formation of **adjectives** (primarily from nouns)

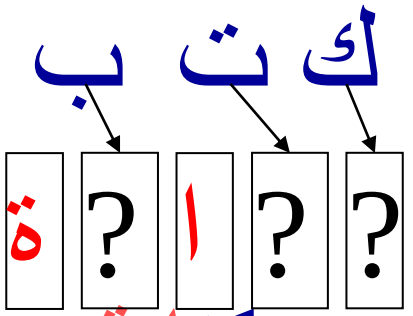
Suffix	Base Noun/Verb	Derived Adjective
-al	computation (N)	computational
-able	embrace (V)	embraceable
-less	clue (N)	clueless

Derivational Morphology



maktoob

written



kitabah

writing

كتاب

book

مكتبة

library

مكتب

office

كتب — ***write***

مكتوب — ***letter***

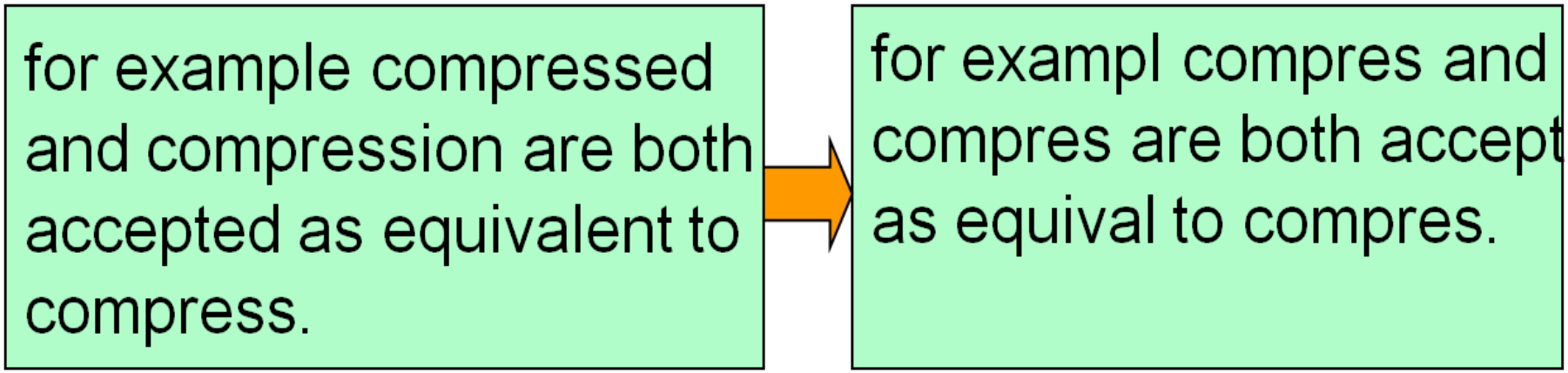
كاتب — ***writer***

Stemming

- Stemming algorithms strip off word affixes yield stem only, no additional information (like plural, 3rd person etc.) used,
  - e.g. in web search engines famous stemming algorithm: the **Porter stemmer**

Stemming

- Reduce tokens to “root” form of words to recognize morphological variation. “computer”, “computational”, “computation” all reduced to same token “compute”
- Correct morphological analysis is language specific and can be complex.
- Stemming “blindly” strips off known affixes (prefixes and suffixes) in an iterative fashion.



Porter Stemmer

- Simple procedure for removing known affixes in English without using a dictionary.
- Can produce unusual stems that are not English words: “computer”, “computational”, “computation” all reduced to same token “comput”
- May conflate (reduce to the same token) words that are actually distinct.
- Does not recognize all morphological derivations
- **Porter stemmer rules**
  - *sses* $\rightarrow$ *ss*
  - *ies* $\rightarrow$ *i*
  - *ational* $\rightarrow$ *ate*
  - *tional* $\rightarrow$ *tion*
  - *ing* $\rightarrow \varepsilon$

Stemming Problems

Errors of Commission		Errors of Omission	
organization	organ	European	Europe
doing	doe	analysis	analyzes
Generalization	Generic	Matrices	matrix
Numerical	numerous	Noise	noisy
Policy	police	sparse	sparsity

Tokenization, Word Segmentation

- Tokenization or word segmentation separate out “words” (lexical entries) from running text expand abbreviated terms
- ***Example:*** *I’m* into *I am*, *it’s* into *it is* collect tokens forming single lexical entry
- ***Example:*** *New York* marked as one single entry
- ***Punctuation*** :
  - ***State-of-the-art***: break up hyphenated sequence. ***U.S.A.*** vs. ***USA***
  - **Numbers**
    - 3/12/91 **Vs** Mar. 12, 1991

Lemmatization

- **Reduce inflectional/derivational forms to base form Direct impact on vocabulary size**
  - ***E.g., am, are, is → be***
  - ***Ex., car, cars, car's, cars' → car***
  - ***the boy's cars are different colors → the boy car be different color***
- ***How to do this?***
  - ***Need a list of grammatical rules + a list of irregular words***
  - ***Children → child, spoken → speak …***
  - ***Practical implementation: use WordNet's morphstr function***

Why parse words?

- For spell-checking
  - Is muncheble a legal word?
- To identify a word's part-of-speech (POS)
  - For **sentence parsing**, for **machine translation**, …
- To identify a word's stem
  - For **information retrieval**
- Why not just list all word forms in a lexicon?

MORPHOLOGICAL PARSING

- **Taking a surface input and identifying its components and underlying structure**
- **Morphological parsing: parsing a word into stem and affixes and identifying the parts and their relationships.**

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`)

Morphological Parser

- In order to build a morphological parser, we'll need at least the following:
  - Lexicon
    - Often not feasible to just list all the words.
    - Some morphological processes are productive.
  - Morphotactics (order of morphemes)
  - Orthographic rules
    - Needed to handle variations of the **spelling** of the stem

Morphological Parser

- In order to build a morphological parser, we'll need at least the following:
  1. a lexicon: The ***list of*** stems and affixes, together with basic information about them (whether a stem is a Noun stem or a Verb stem, etc).
  2. morphotactics: the model of ***morpheme ordering*** that explains which classes of morphemes can follow other classes of morphemes inside a word.
  3. orthographic rules: these ***spelling rules*** are used to model the changes that occur in a word,
     - E.g. English nouns ending in *–y* change to *–i* (city → cities)

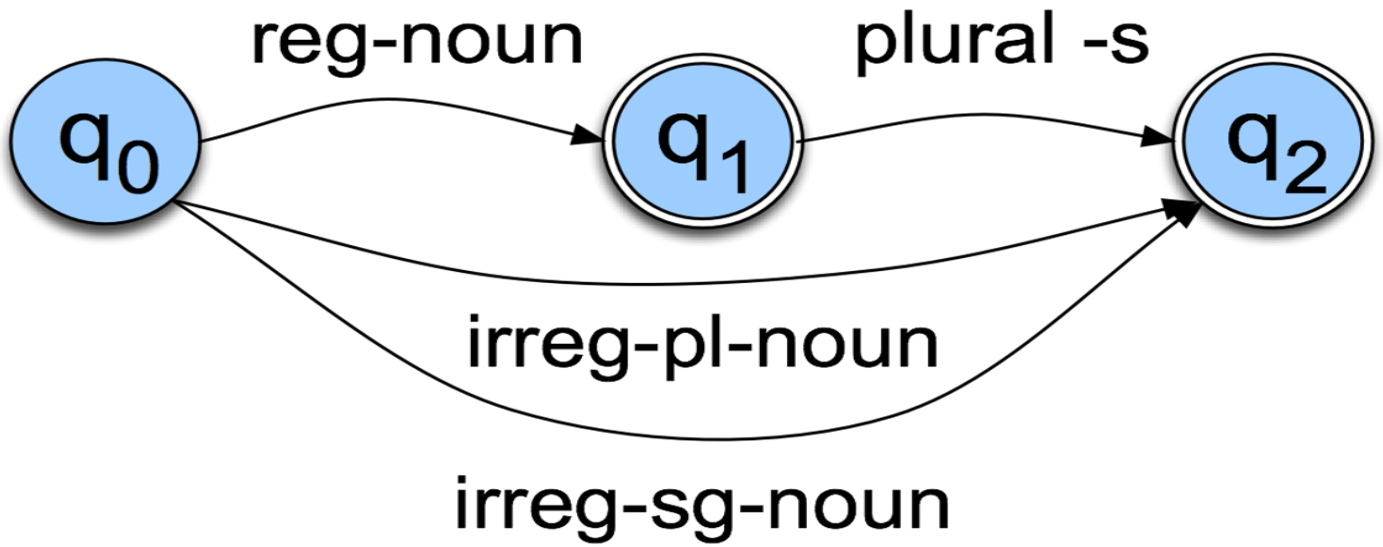
Lexicons

- A lexicon is a repository for words.
  - The simplest possible lexicon would consist of an explicit list of every word of the language (every word)

- Lexicon can be stored as an FSA.
- A base lexicon (with baseforms) can be plugged into a larger FSA to capture morphological rules and morphotactics.

Finite State Automaton

- Regular singular nouns are ok
- Regular plural nouns have an -s on the end
- Irregulars are ok as is (i.e. treat as atomic for now)
- **There are many ways to model morphotactics;**
  - **the finite-state automaton.**
  - **A very simple finite-state model for English nominal inflection.**

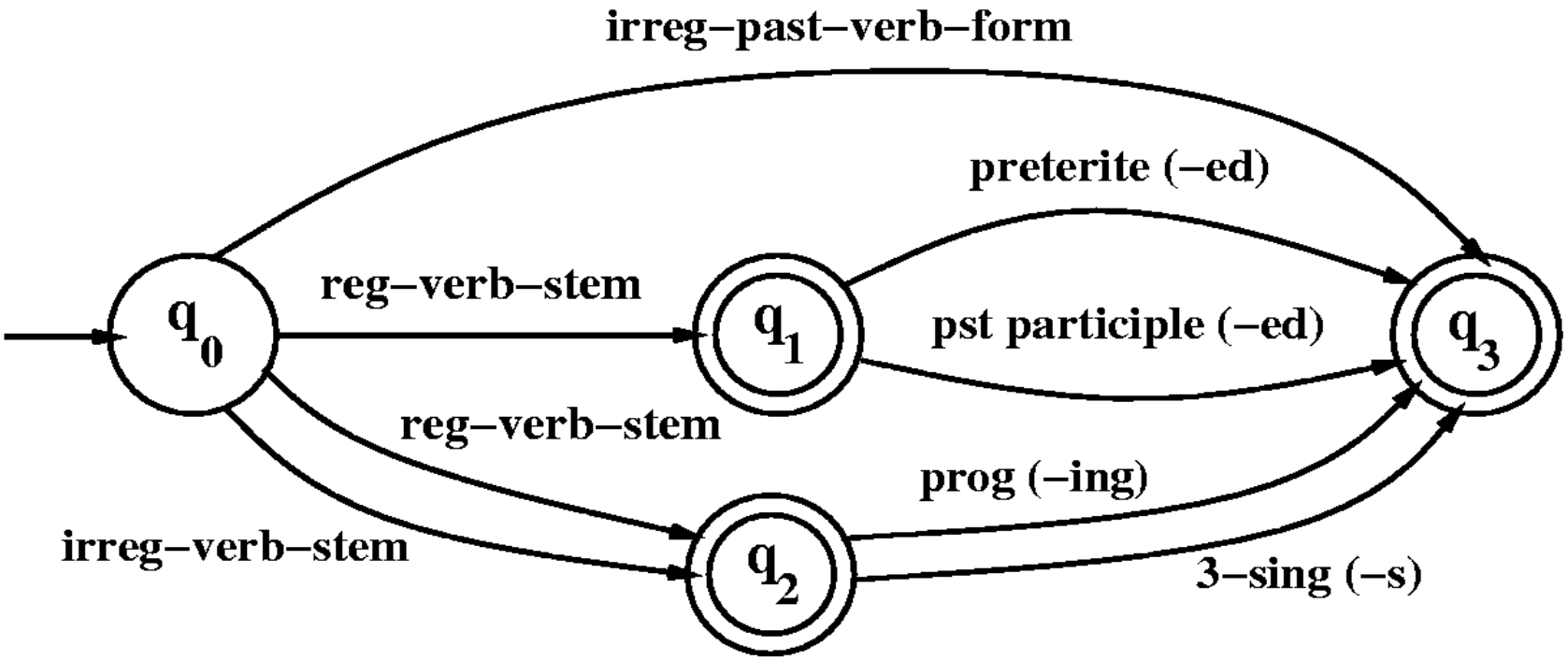


English nominal inflection

reg-noun	irreg-pl-noun	irreg-sg-noun	plural
fox	geese	goose	-s
cat	sheep	sheep	
dog	mice	mouse	
aardvark			

Finite State Automaton

- A finite-state automaton for English verbal inflection

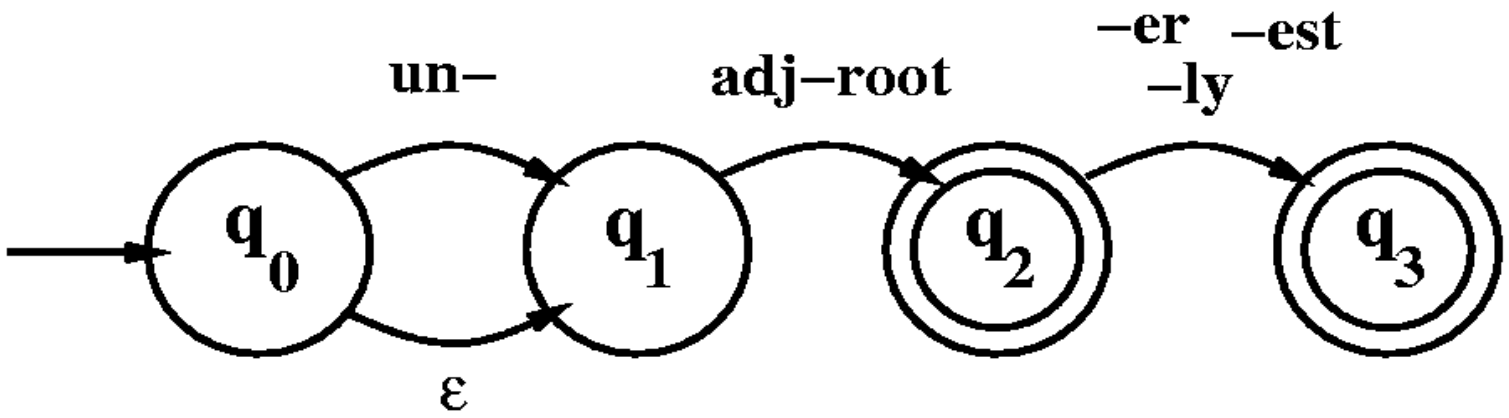


English verbal inflection

reg-verb-stem	irreg-verb-stem	irreg-past-verb	past	past-part	pres-part	3sg
walk	cut	caught	-ed	-ed	-ing	-s
fry	speak	ate				
talk	sing	eaten				
impeach	sang					
	cut					
	spoken					

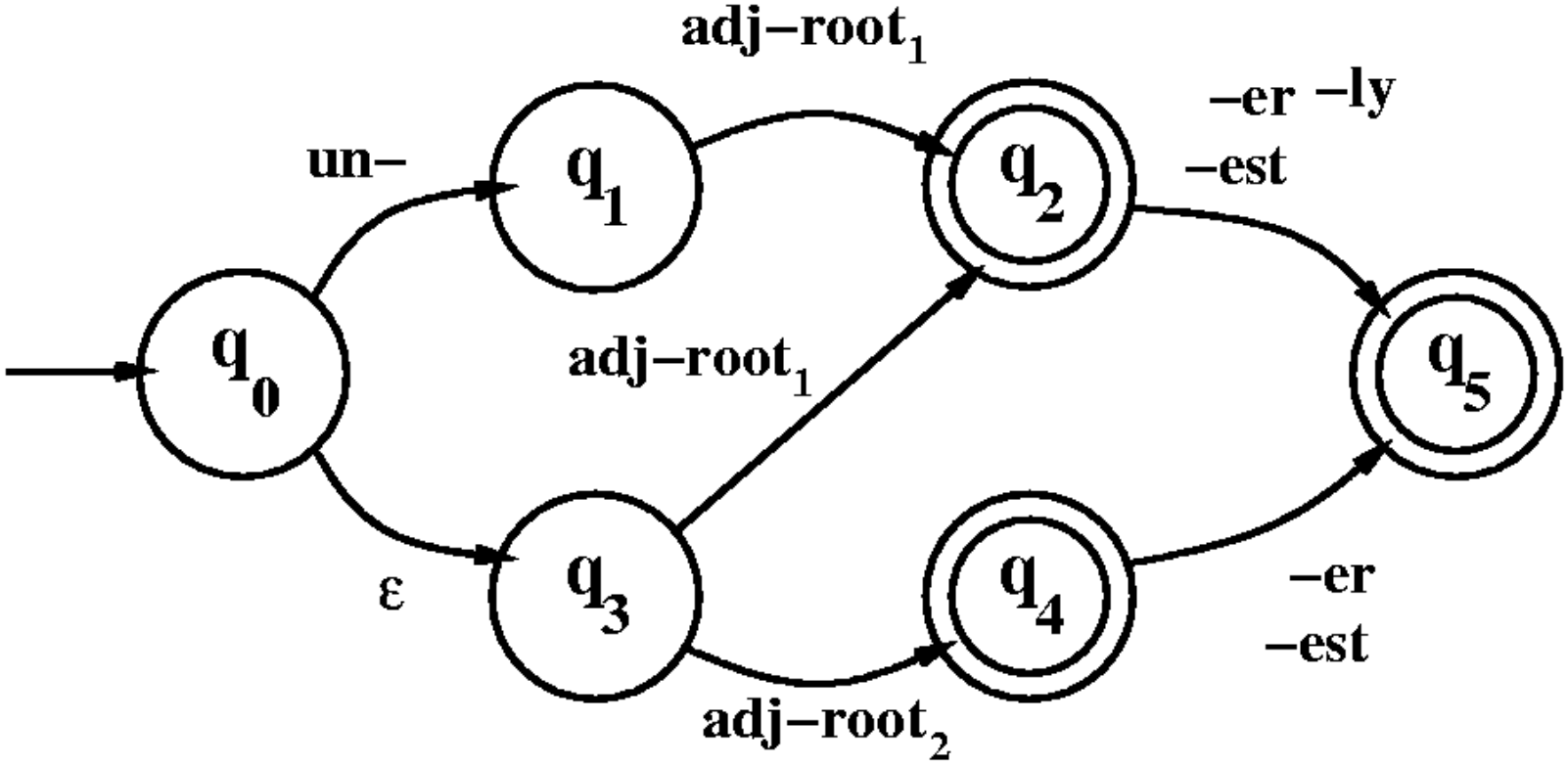
English adjective morphology

- **An FSA for a fragment of English adjective morphology:**



- **So adj-root would include adjectives that can occur with un- and -ly (clear, happy, and real)**

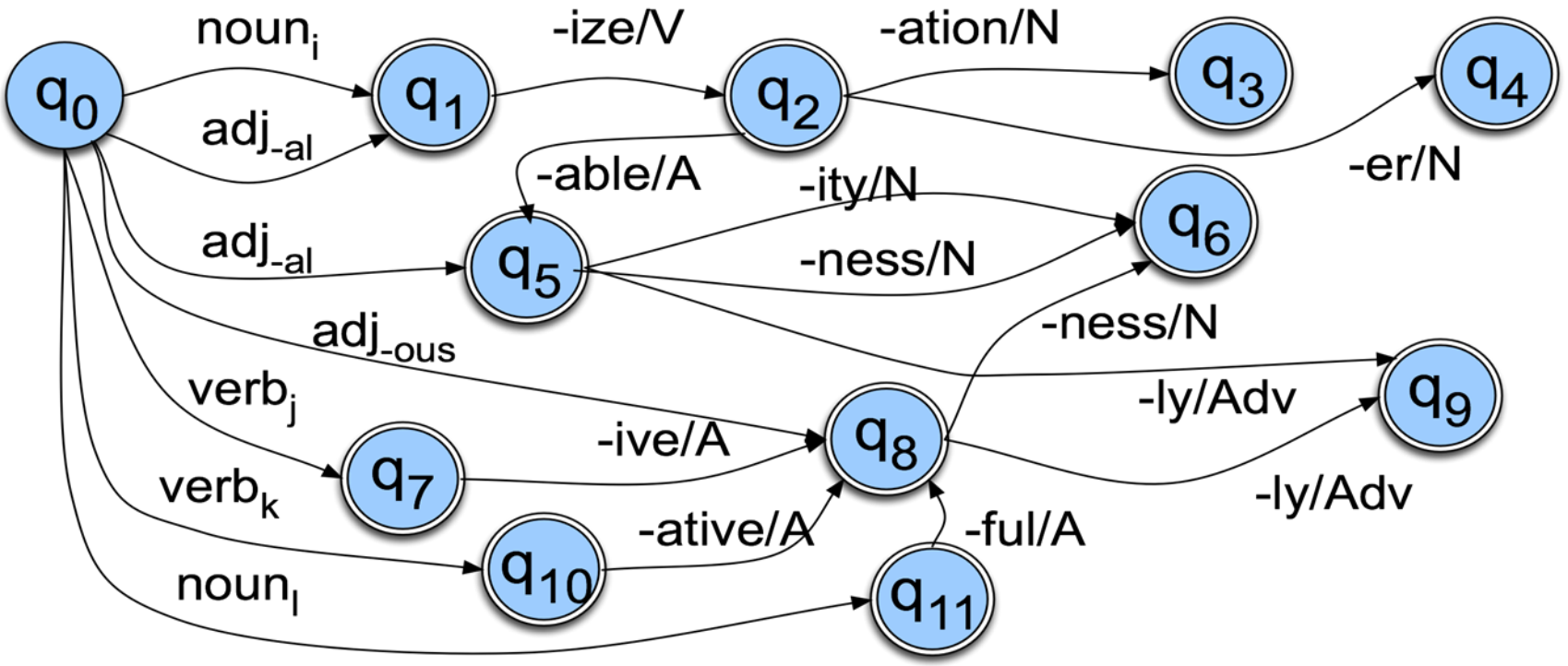
English adjective morphology



- Adj-root$_1$: clear, happy, real
- Adj-root$_2$: big, red, cool

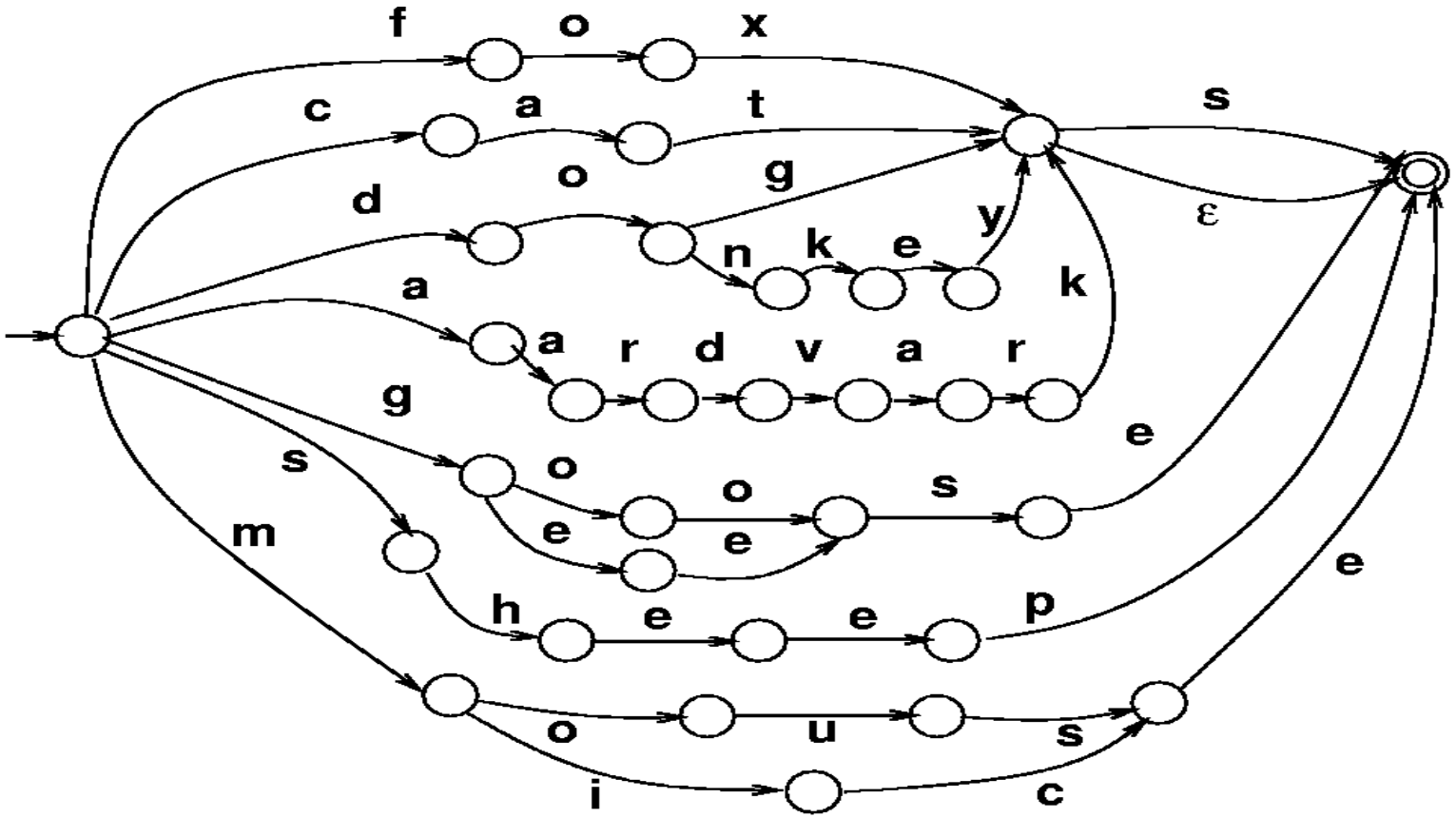
Derivational Rules

- An FSA for another fragment of English derivational morphology.
- If everything is an accept state how do things ever get rejected?
- These FSAs to solve the problem of morphological recognition; that is, of determining whether an input string of letters makes up a legitimate English word or not.
  - We do this by taking the morphotactic



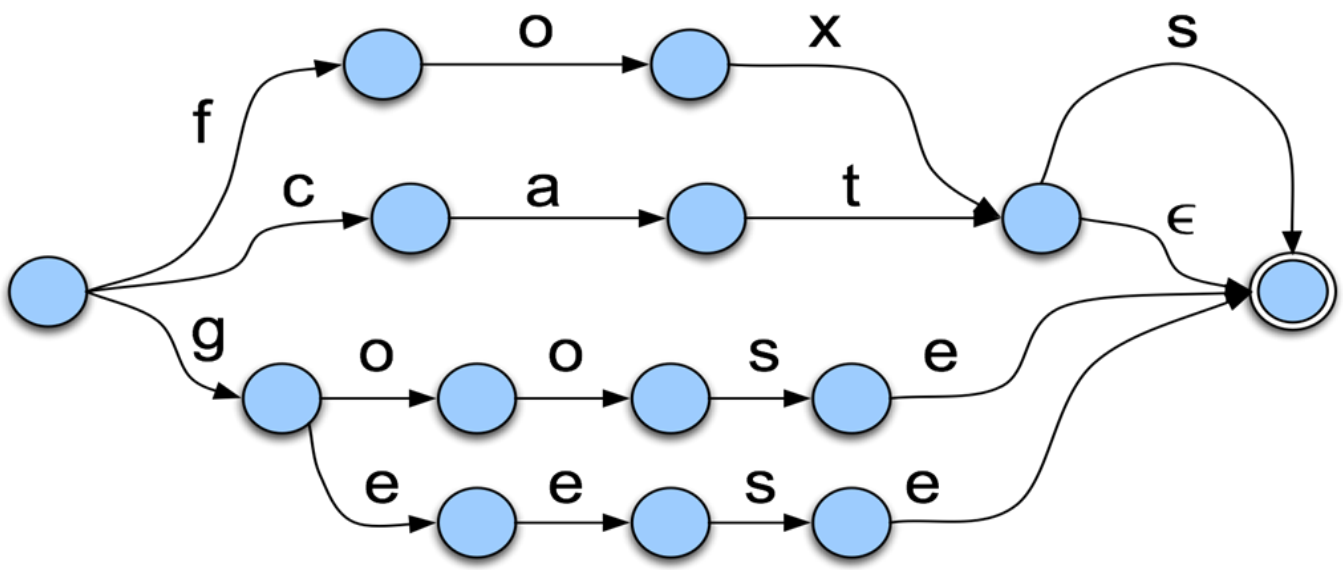
Finite State Automaton

- **We can now use these FSAs to solve the problem of morphological recognition; that is, of determining whether an input string of letters makes up a legitimate English word or not.**



Substitute words for word classes

- Idea is to be able to use this kind of FSA for recognition.
- We've replaced classes like "reg-noun" with the actual words.



Morphological Parsing

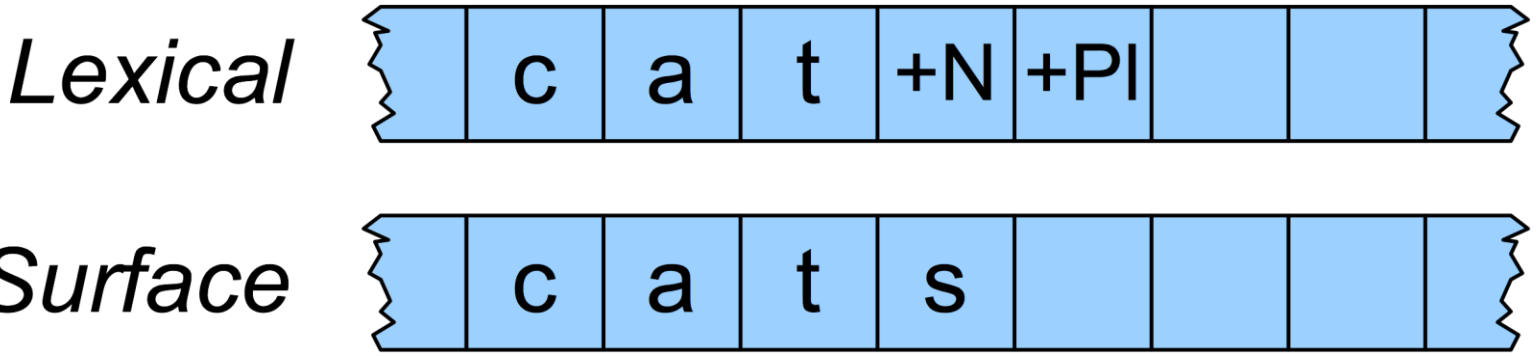
- Use FSAs to represent the lexicon and incidentally do morphological recognition.
- A transducer is a machine that takes input of a certain form and outputs something of a different form.
  - Add another tape
  - Add extra symbols to the transitions
    - Example: On one tape we read "cats", on the other we write "cat +N +PL" telling us that cat is a plural noun.

Two level morphology

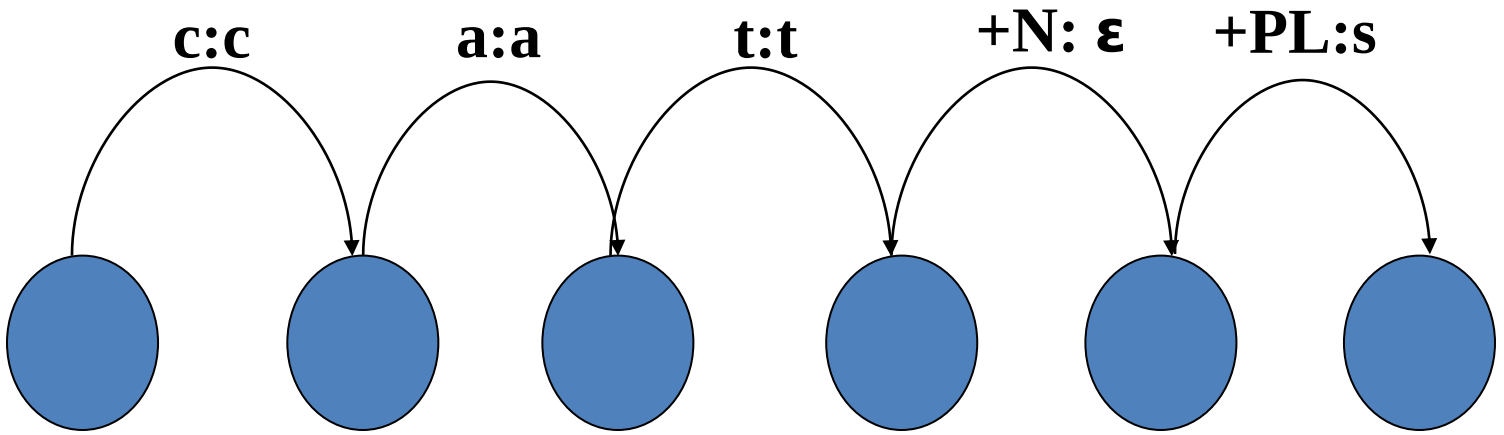
- We will do this via a version of two-level morphology, first proposed by Koskenniemi (1983).
- Two level morphology represents a word as a correspondence between
  - ***lexical level***, which represents a simple concatenation of morphemes making up a word,
  - ***surface level***, which represents the actual spelling of the final word.

Morphological Parsing

- Morphological parsing is implemented by building mapping rules that map letter sequences
- ***surface level*** →actual spelling like cats
- ***lexical level*** → stem for a word **+** *morphological information* (+N +PL)



Transitions



- **c:c** means read a c on one tape and write a c on the other
- **+N:ε** means read a +N symbol on one tape and write nothing on the other
- **+PL:s** means read +PL and write an s
- *Note the conventions:* x:y represents an input symbol x and the output symbol y.

Finite-State Transducers

- The automaton that we use for performing the mapping between these two levels is the finite-state transducer or FST.
- A transducer maps between one set of symbols and another;
- FST as a two-tape automaton which recognizes or generates pairs of strings