Structure of Words

Morphology

What is a word?

- How many words do you find in the following short text?

What problems do you encounter?

It's a shame that our data-base is not up-to-date. It is a shame that um, data base A costs $2300.50 and that database B costs $5000. All databases cost far too much.

- Time: 3 minutes

Counting words: tokenization

- **Tokenization** is a processing step where the input text is
- automatically divided into units called **tokens** where each is either a **word** or a number or a punctuation mark…
- So, word count can ignore numbers, punctuation marks (?)

COUNTING WORDS

- **Word:** Continuous alphanumeric characters delineated by whitespace.

- **Whitespace:** space, tab, newline.

- BUT dividing at spaces is too simple: **It's, data base**

Basic Text Processing

Regular Expressions

Regular expressions

A formal language for specifying text strings

How can we search for any of these?

- woodchuck
- woodchucks
- Woodchuck
- Woodchucks

Regular Expressions: Disjunctions

Letters inside square brackets []

Pattern	Matches
[wW]oodchuck	Woodchuck, woodchuck
[1234567890]	Any digit

Ranges [A-Z]

Pattern	Matches	
[A-Z]	An upper case letter	Drenched Blossoms
[a-z]	A lower case letter	my beans were impatient
[0-9]	A single digit	Chapter 1: Down the Rabbit Hole

Regular Expressions: Negation in Disjunction

[^Ss]

Carat means negation only when first in []

Pattern	Matches	
[^A-Z]	Not an upper case letter	Oyfn pripetchik
[^Ss]	Neither 'S' nor 's'	I have no exquisite reason"
[^e^]	Neither e nor ^	Look here
a^b	The pattern a carat b	Look up a^b now

Regular Expressions: More Disjunction

!

The pipe | for disjunction

Pattern	Matches
groundhog|woodchuck	
yours|mine	yours mine
a|b|c	= [abc]
[gG]roundhog|[Ww]oodchuck	

Regular Expressions: ? * + .

Pattern	Matches	
colou?r	Optional previous char	color colour
oo*h!	0 or more of previous char	oh! ooh! oooh! ooooh!
o+h!	1 or more of previous char	oh! ooh! oooh! ooooh!
baa+		baa baaa baaaa baaaaa
beg.n		begin begun begun beg3n

Stephen C Kleen

Kleene *, Kleene

Search for Some Tokenization Issues

Identify at least 3

Some Tokenization Issues

- Sentence Boundaries
  - Punctuation, eg quotation marks around sentences?
  - Periods – end of line or not?

Some Tokenization Issues

- Proper Names
- What to do about
  - “New York-New Jersey train”?
  - “California Governor Arnold Schwarzenegger”?

Some Tokenization Issues

- Contractions
  - That’s Fred’s jacket’s pocket.
  - I’m doing what you’re saying “Don’t do!”.

JABBERWOCKY

Lewis Carroll

(from *Through the Looking-Glass and What Alice Found There*, 1872)

`Twas brillig, and the slithy toves
Did gyre and gimble in the wabe:
All mimsy were the borogoves,
And the mome raths outgrabe.

"Beware the Jabberwock, my son!
The jaws that bite, the claws that catch!
Beware the Jubjub bird, and shun
The frumious Bandersnatch!"

He took his vorpal sword in hand:
Long time the manxome foe he sought --
So rested he by the Tumtum tree,
And stood awhile in thought.

And, as in uffish thought he stood,
  The Jabberwock, with eyes of flame,
Came whiffling through the tulgey wood,
  And burbled as it came!

One, two! One, two! And through and through
  The vorpal blade went snicker-snack!
He left it dead, and with its head
  He went galumphing back.

"And, has thou slain the Jabberwock?
  Come to my arms, my beamish boy!
O frabjous day! Callooh! Callay!'
  He chortled in his joy.

`Twas brillig, and the slithy toves
  Did gyre and gimble in the wabe;
All mimsy were the borogoves,
  And the mome raths outgrabe.

What do you think are the meaning of the following words?

- Chortled
- Galumphing
- Toves
- Wabe

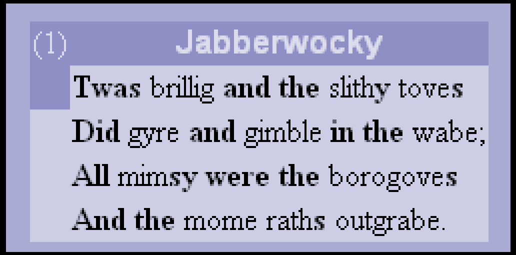
Jabberwocky Analysis

- Why do we pretty much understand the words?

- We recognize combinations of morphemes/words.
  - **Chortled** - Laugh in a breathy, gleeful way; (Definition from Oxford American Dictionary) A combination of "chuckle" and "snort."
  - **Galumphing** - Moving in a clumsy, ponderous, or noisy manner. Perhaps a blend of "gallop" and "triumph." (Definition from Oxford American Dictionary)

Jabberwocky Analysis

- Why do we pretty much understand the words?
  - Surrounding English words strongly indicate the parts-of-speech of the nonsense words.



(1) **Jabberwocky**

Twas brillig and the slithy toves
Did gyre and gimble in the wabe;
All mimsy were the borogoves
And the mome raths outgrabe.

  - ***toves***: probably can perform an action

(because they **did *gyre*** and ***gimble***)

  - ***wabe***: is probably a place.

(they **did** … **in the *wabe***)

http://assets.cambridge.org/052185/542X/excerpt/052185542X_excerpt.pdf

Jabberwocky Analysis

- Surrounding English words strongly indicate the parts-of-speech of the nonsense words.

(1) **Jabberwocky**

Twas brillig **and the** slithy toves
Did gyre **and** gimble **in the** wabe;
All mimsy **were the** borogoves
And the mome raths outgrabe.

Example from http://www.departments.bucknell.edu/linguistics/lectures/05lect02.html

Morphology

- Morphology:
  - The study of the way words are built up from smaller meaning units.
- Morphemes:
  - The smallest meaningful unit in the grammar of a language.

Modified from Dorr and Habash (after Jurafsky and Martin)

Morphology

- Contrasts:
  - Derivational *vs.* Inflectional
  - Regular *vs.* Irregular
  - Concatinative *vs.* Templatic (root-and-pattern)

Morphemes and Words

Combine morphemes to create words

Inflection

- combination of a word stem with a grammatical morpheme
- same word class, e.g. kain (verb), kumain (verb)

Derivation

- combination of a word stem with a grammatical morpheme
- Yields different word class, e.g. kanta (noun), kumanta (verb)

Compounding

- combination of multiple word stems, e.g. bahay-kubo

Cliticization

- combination of a word stem with a clitic
- different words from different syntactic categories, e.g. I've = I + have

Inflectional Morphology (verbs)

Verb Inflections for:

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

Morpholog. Form Regularly Inflected Form
stem walk mergetry map

-s form walks merges tries maps

-ing participle walking merging trying mapping

past; -ed participlewalked 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 (nouns)

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

Inflectional and Derivational Morphology (adjectives)

Adjective Inflections and Derivations:

prefix un- unhappy adjective, negation

suffix -ly happily adverb, mode

-er happier adjective, comparative 1

-est happiest adjective, comparative 2

suffix -ness happiness noun

plus combinations, like unhappiest, unhappiness.

Distinguish different adjective classes, which can or cannot take certain inflectional or derivational forms, e.g. no negation for big.

Inflectional Morphology

Inflectional Morphology

word stem + grammatical morpheme   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)

Derivational Morphology (nouns)

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

Derivational Morphology (adjectives)

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

Methods, Algorithms

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.

for example compressed and compression are both accepted as equivalent to compress.

→

for exampl compres and compres are both accept as equival to compres.

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

Typical rules in Porter stemmer

$sses \rightarrow ss$

$ies \rightarrow i$

$ational \rightarrow ate$

$tional \rightarrow tion$ $\quad$ $ing \rightarrow \varepsilon$

Stemming Problems

Errors of Comission		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

E.g. *I’m* into *I am*, *it’s* into *it is*

collect tokens forming single lexical entry

E.g. *New York* marked as one single entry

Simple Tokenization

Analyze text into a sequence of discrete tokens (words).

Sometimes punctuation (e-mail), numbers (1999), and case (Republican vs. republican) can be a meaningful part of a token.

- However, frequently they are not.

Simplest approach is to ignore all numbers and punctuation and use only case-insensitive unbroken strings of alphabetic characters as tokens.

More careful approach:

- Separate ? ! ; : “ ‘ [ ] ( ) < >
- Care with . - why? when?
- Care with … ??

Punctuation

Children's: use language-specific mappings to normalize (e.g. Anglo-Saxon genitive of nouns, verb contractions: won't -> wo 'nt)

State-of-the-art: break up hyphenated sequence.

U.S.A. vs. ***USA***

a.out

Numbers

3/12/91

Mar. 12, 1991

55 B.C.

B-52

100.2.86.144

- Generally, don't index as text
- Creation dates for docs

Lemmatization

Reduce inflectional/derivational forms to base form

Direct impact on vocabulary size

E.g.,

am, are, is → be

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 $\rightarrow$ child, spoken $\rightarrow$ speak …

Practical implementation: use WordNet's morphstr function

Perl: WordNet::QueryData (first returned value from validForms function)

Morphological Processing

Knowledge

- lexical entry: stem plus possible prefixes, suffixes plus word classes, e.g. endings for verb forms (see tables above)
- rules: how to combine stem and affixes, e.g. add s to form plural of noun as in dogs
- orthographic rules: spelling, e.g. double consonant as in mapping

Processing: Finite State Transducers

- take information above and analyze word token / generate word form

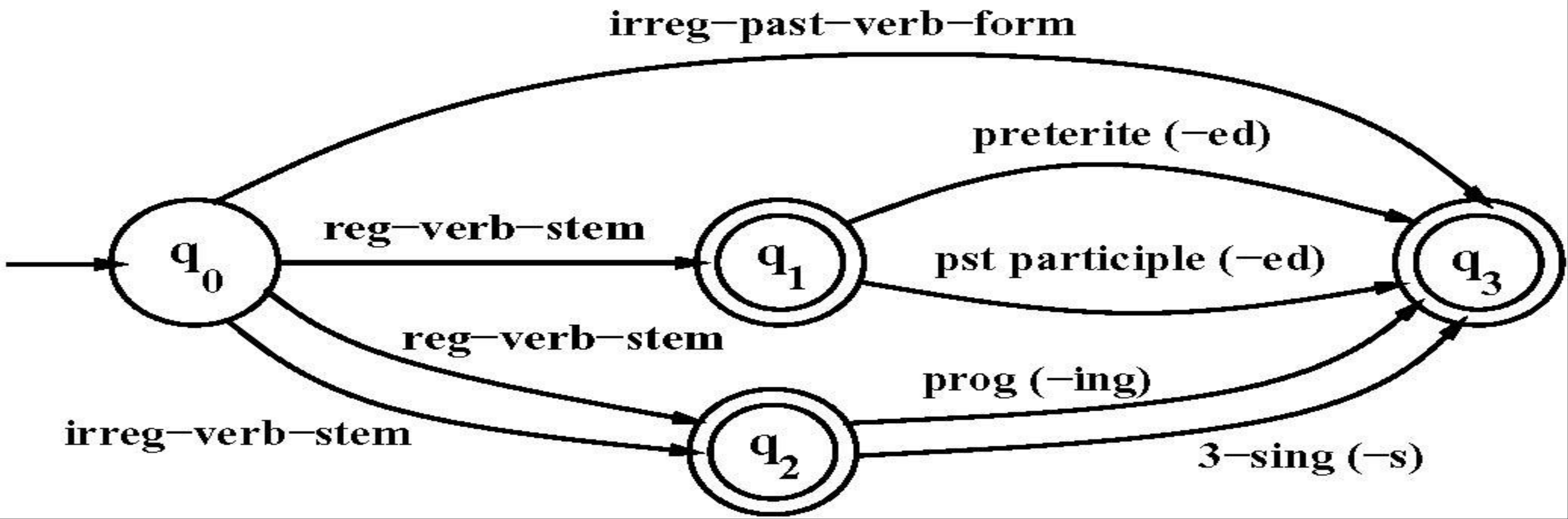


Fig. 3.3 FSA for verb inflection.

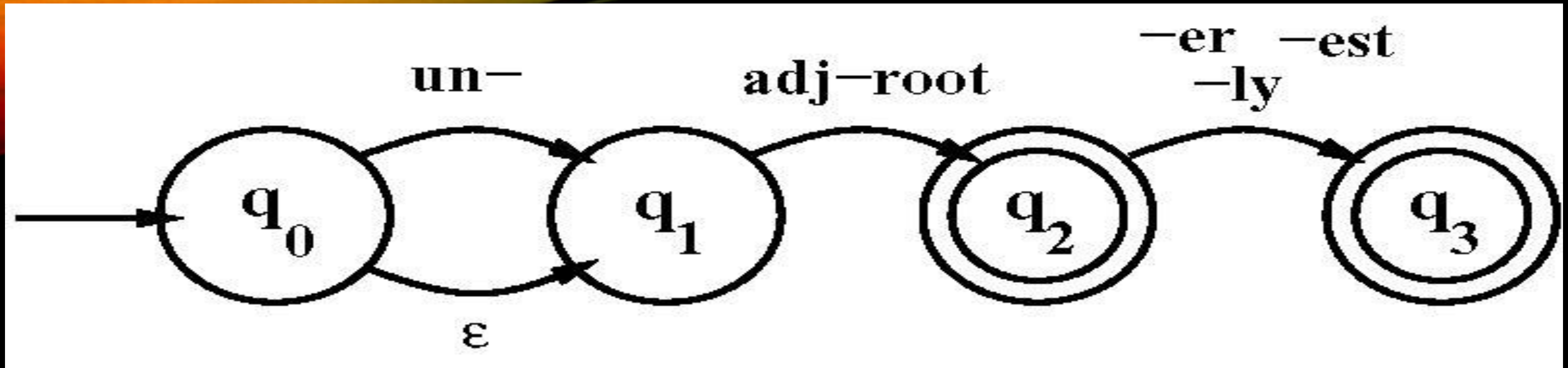


Fig. 3.4 Simple FSA for adjective inflection.

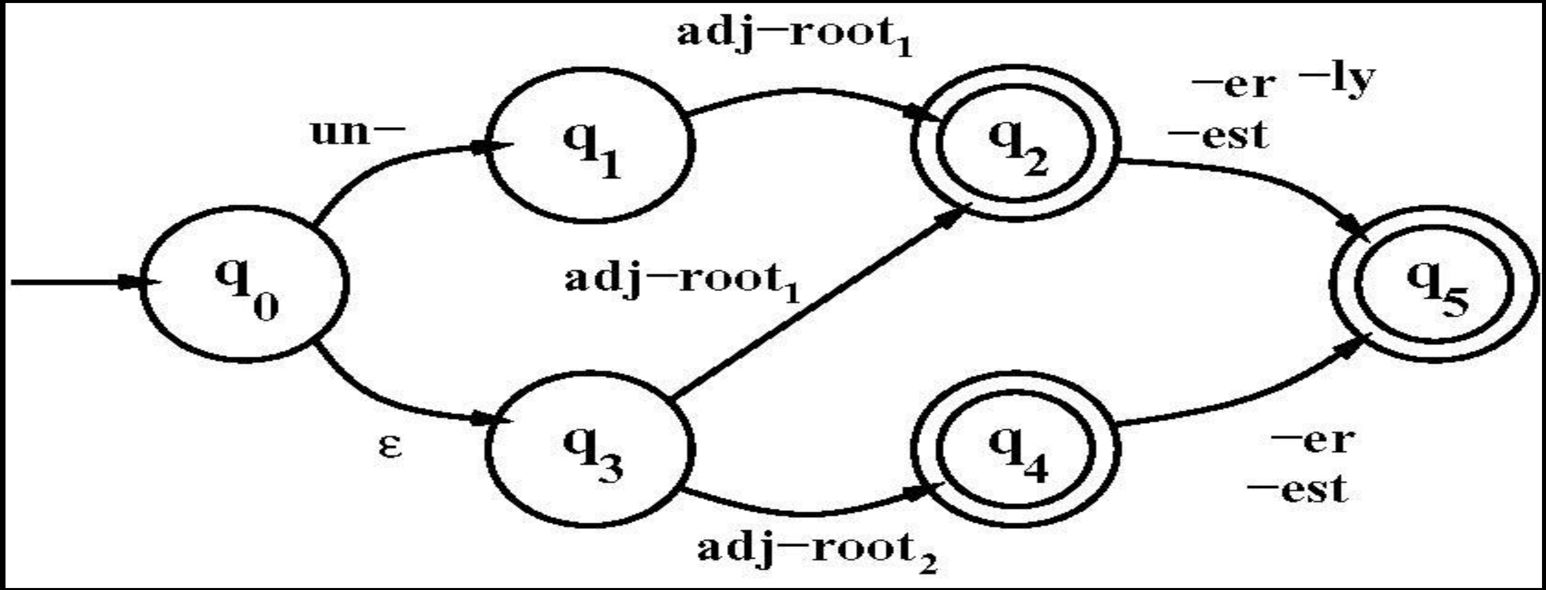


Fig. 3.5 More detailed FSA for adjective inflection.

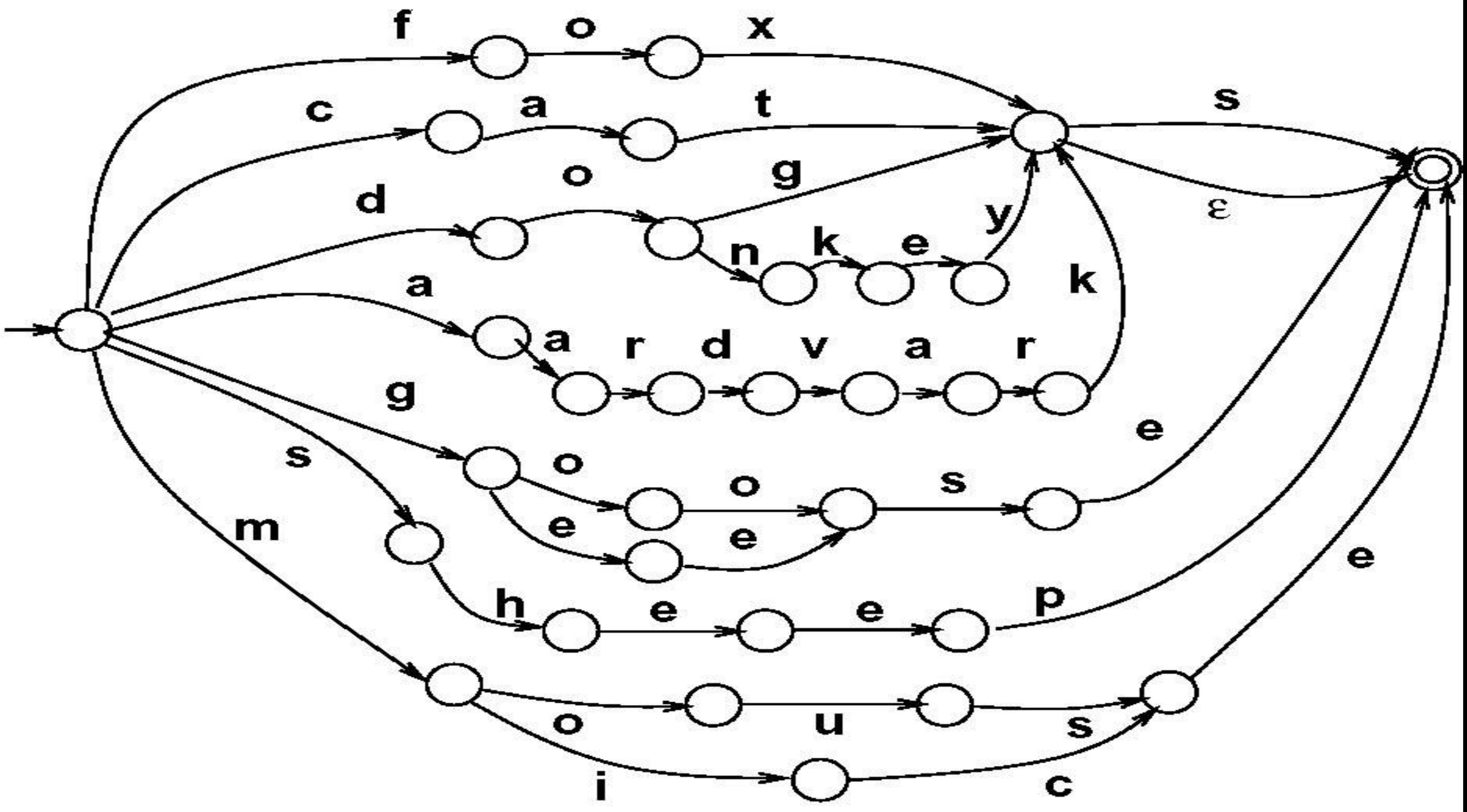


Fig. 3.7 Compiled FSA for noun inflection.

Basic Text Processing

Sentence Segmentation and Decision Trees

Sentence Segmentation

!, ? are relatively unambiguous

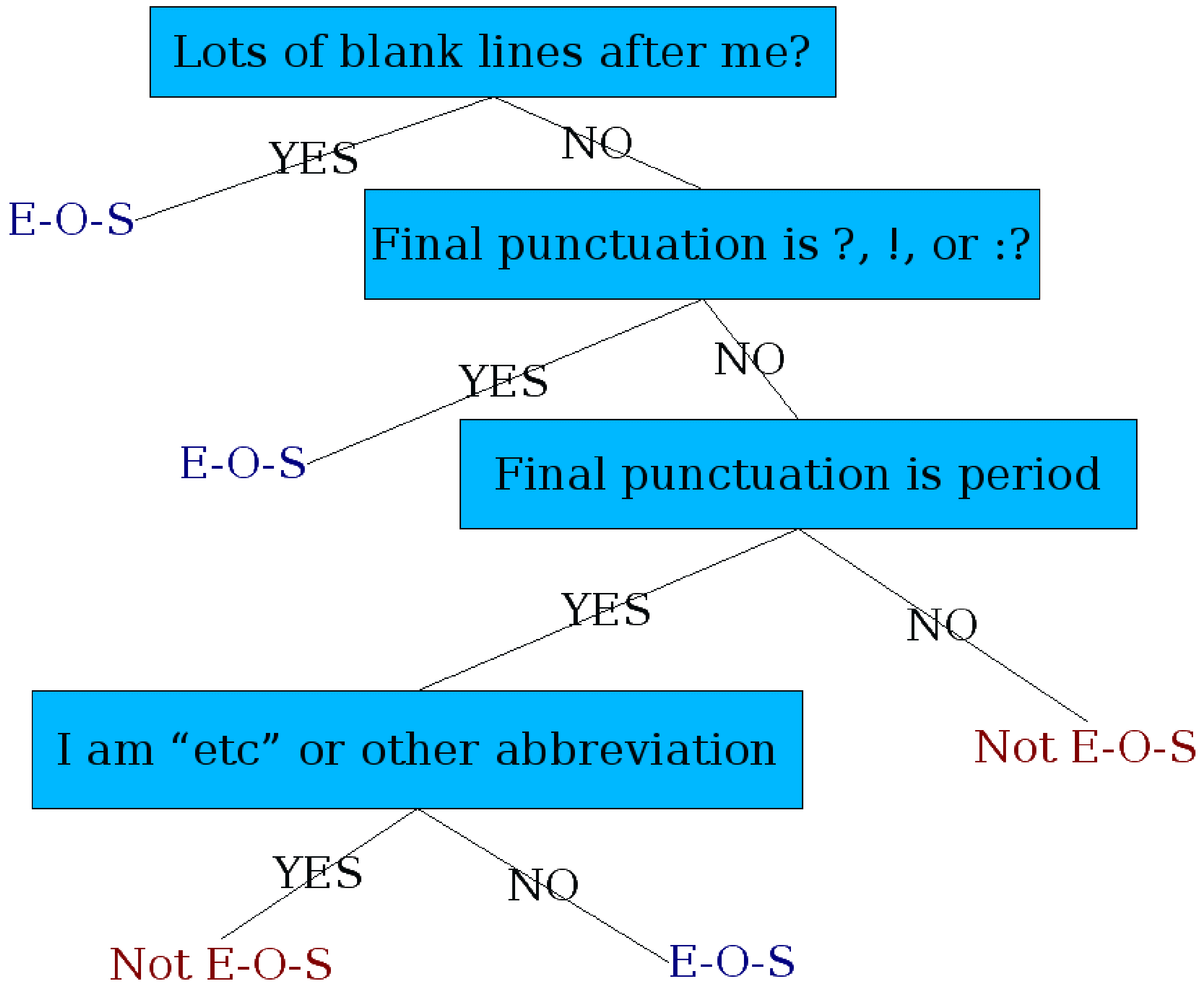
Period "." is quite ambiguous

- Sentence boundary
- Abbreviations like Inc. or Dr.
- Numbers like .02% or 4.3

Build a binary classifier

- Looks at a "."
- Decides EndOfSentence/NotEndOfSentence
- Classifiers: hand-written rules, regular expressions, or machine-learning

Determining if a word is end-of-sentence: a Decision Tree



More sophisticated decision tree features

- Case of word with ".": Upper, Lower, Cap, Number
- Case of word after ".": Upper, Lower, Cap, Number
- Numeric features
  - Length of word with "."
  - Probability(word with "." occurs at end-of-s)
  - Probability(word after "." occurs at beginning-of-s)

Implementing Decision Trees

- A decision tree is just an if-then-else statement
- The interesting research is choosing the features
- Setting up the structure is often too hard to do by hand
  - Hand-building only possible for very simple features, domains
    - For numeric features, it's too hard to pick each threshold
  - Instead, structure usually learned by machine learning from a training corpus

Decision Trees and other classifiers

We can think of the questions in a decision tree

As features that could be exploited by any kind of classifier

- Logistic regression
- SVM
- Neural Nets
- etc.

Assignment

Identify at most 10 Morphological Challenges within the Filipino Language.

Work with a pair.

To be submitted in a text file via eleap assignment.