Morphology and Finite-state Transducers Part 2

ICS 482: Natural Language Processing

Lecture 6

Husni Al-Muhtaseb

بسم الله الرحمن الرحيم

ICS 482: Natural Language Processing

Lecture 6

Morphology and Finite-state Transducers Part 2

Husni Al-Muhtaseb

NLP Credits and Acknowledgment

These slides were adapted from presentations of the Authors of the book

SPEECH and LANGUAGE PROCESSING:

An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition

and some modifications from presentations found in the WEB by several scholars including the following

NLP Credits and Acknowledgment

If your name is missing please contact me
muhtaseb
At
Kfupm.
Edu.
sa

NLP Credits and Acknowledgment

Husni Al-Muhtaseb
James Martin
Jim Martin
Dan Jurafsky
Sandiway Fong
Song young in
Paula Matuszek
Mary-Angela Papalaskari
Dick Crouch
Tracy Kin
L. Venkata Subramaniam
Martin Volk
Bruce R. Maxim
Jan Hajič
Srinath Srinivasa
Simeon Ntafos
Paolo Pirjanian
Ricardo Vilalta
Tom Lenaerts

Heshaam Feili
Björn Gambäck
Christian Korthals
Thomas G. Dietterich
Devika Subramanian
Duminda Wijesekera
Lee McCluskey
David J. Kriegman
Kathleen McKeown
Michael J. Ciaraldi
David Finkel
Min-Yen Kan
Andreas Geyer-Schulz
Franz J. Kurfess
Tim Finin
Nadjet Bouayad
Kathy McCoy
Hans Uszkoreit

Khurshid Ahmad
Staffan Larsson
Robert Wilensky
Feiyu Xu
Jakub Piskorski
Rohini Srihari
Mark Sanderson
Andrew Elks
Marc Davis
Ray Larson
Jimmy Lin
Marti Hearst
Andrew McCallum
Nick Kushmerick
Mark Craven
Chia-Hui Chang
Diana Maynard
James Allan

Martha Palmer
julia hirschberg
Elaine Rich
Christof Monz
Bonnie J. Dorr
Nizar Habash
Massimo Poesio
David Goss-Grubbs
Thomas K Harris
John Hutchins
Alexandros Potamianos
Mike Rosner
Latifa Al-Sulaiti
Giorgio Satta
Jerry R. Hobbs
Christopher Manning
Hinrich Schütze
Alexander Gelbukh
Gina-Anne Levow
Guitao Gao
Qing Ma
Zeynep Altan

Previous Lectures

- **1 Pre-start questionnaire**
- **2 Introduction and Phases of an NLP system**
- **2 NLP Applications**
- **3 Chatting with Alice**
- **3 Regular Expressions, Finite State Automata**
- **3 Regular languages**
- **4 Regular Expressions & Regular languages**
- **4 Deterministic & Non-deterministic FSAs**
- **5 Morphology: Inflectional & Derivational**
- **5 Parsing**

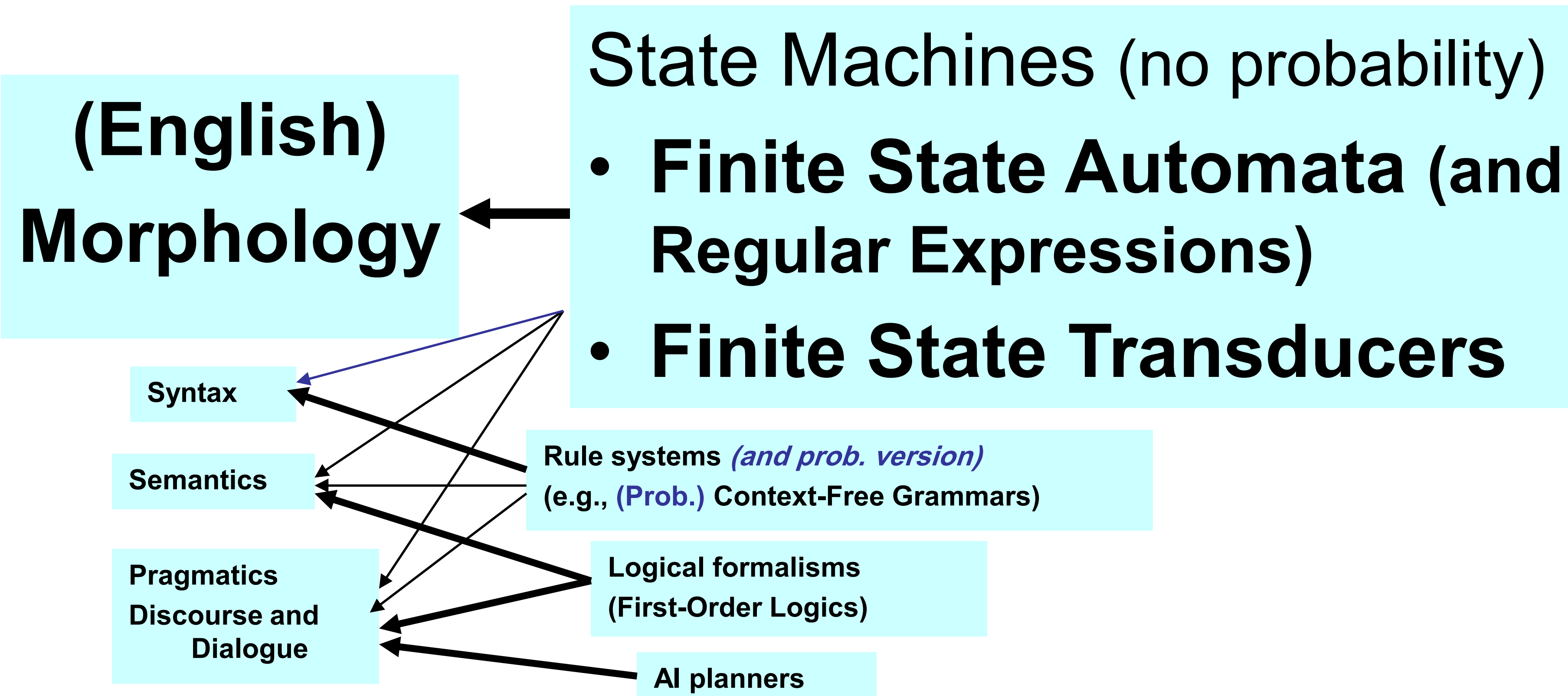
Today's Lecture

- **Review of Morphology**
- **Finite State Transducers**
- **Stemming & Porter Stemmer**

Reminder: Quiz 1 Next class

- **Next time: Quiz**
  - **Ch 1!, 2, & 3 (Lecture presentations)**
  - **Do you need a sample quiz?**
    - **What is the difference between a sample and a template?**
    - **Let me think – It might appear at the WebCt site on late Saturday.**

Introduction



English Morphology

- **Morphology is the study of the ways that words are built up from smaller meaningful units called morphemes**
- **morpheme classes**
  - **Stems: The core meaning bearing units**
  - **Affixes: Adhere to stems to change their meanings and grammatical functions**
  - **Example: *unhappily***

English Morphology

- **We can also divide morphology up into two broad classes**
  - **Inflectional**
  - **Derivational**

- **Non English**
  - **Concatinative Morphology**
  - **Templatic Morphology**

Word Classes

- **By word class, we have in mind familiar notions like noun, verb, adjective and adverb**
- **Why to concerned with word classes?**
  - **The way that stems and affixes combine is based to a large degree on the word class of the stem**

Inflectional Morphology

- **Word building process that serves grammatical function without changing the part of speech or the meaning of the stem**
- **The resulting word**
  - **Has the same word class as the original**
  - **Serves a grammatical/ semantic purpose different from the original**

Inflectional Morphology in English

on Nouns

- PLURAL ***-s*** *book**s***
- POSSESSIVE ***-'s*** *Mary**'s***

on Verbs

- 3 SINGULAR ***-s*** *s/he know**s***
- PAST TENSE ***-ed*** *talk**ed***
- PROGRESSIVE ***-ing*** *talk**ing***
- PAST PARTICIPLE ***-en, -ed*** *writt**en**, talk**ed***

on Adjectives

- COMPARATIVE ***-er*** *long**er***
- SUPERLATIVE ***-est*** *long**est***

Nouns and Verbs (English)

- **Nouns are simple**
  - **Markers for plural and possessive**
- **Verbs are slightly more complex**
  - **Markers appropriate to the tense of the verb**
- **Adjectives**
  - **Markers for comparative and superlative**

Regulars and Irregulars

- **some words misbehave (refuse to follow the rules)**
  - **Mouse/mice, goose/geese, ox/oxen**
  - **Go/went, fly/flew**
- **The terms regular and irregular will be used to refer to words that follow the rules and those that don't.**

Regular and Irregular Verbs

- **Regulars…**
  - **Walk, walks, walking, walked, walked**
- **Irregulars**
  - **Eat, eats, eating, ate, eaten**
  - **Catch, catches, catching, caught, caught**
  - **Cut, cuts, cutting, cut, cut**

Derivational Morphology

- **word building process that creates new words, either by changing the meaning or changing the part of speech of the stem**
  - **Irregular meaning change**
  - **Changes of word class**

Examples of derivational morphemes in English that change the part of speech

- *ful* (N → Adj)
  - ***pain*** → ***pain****ful*
  - ***beauty*** → ***beauti****ful*
  - ***truth*** → ***truth****ful*
  - ***cat*** → ****cat****ful*
  - ***rain*** → ****rain****ful*
- *ment* (V → N)

  establish → ***establish****ment*
- *ity* (Adj → N)
  - ***pure*** → ***pur****ity*
- *ly* (Adj → Adv)
  - ***quick*** → ***quick****ly*
- *en* (Adj → V)
  - ***wide*** → ***wid****en*

Examples of derivational morphemes in English that change the meaning

- *dis-*
  - ***appear*** → *dis**appear***
- *un-*
  - ***comfortable*** → *un**comfortable***
- *in-*
  - ***accurate*** → *in**accurate***
- *re-*
  - ***generate*** → *re**generate***
- *inter-*
  - ***act*** → *inter**act***

Examples on Derivational Morphology

V → N

compute	**computer**
nominate	**nominee**
deport	**deportation**
computerize	**computerization**

N → V

computer	**computerize**

A → N

furry	**furriness**
apt	**aptitude**
sincere	**sincerity**

N → A

cat	**catty, catlike**
hope	**hopeless**
magic	**magical**

V → A

love	**lovable**

A → V

black	**blacken**
modern	**modernize**

Derivational Examples

- **Verb/Adj to Noun**

-ation	**computerize**	**computerization**
-ee	**appoint**	**appointee**
-er	**kill**	**killer**
-ness	**fuzzy**	**fuzziness**

Derivational Examples

- **Noun/ Verb to Adj**

-al	**Computation**	**Computational**
-able	**Embrace**	**Embraceable**
-less	**Clue**	**Clueless**

Compute

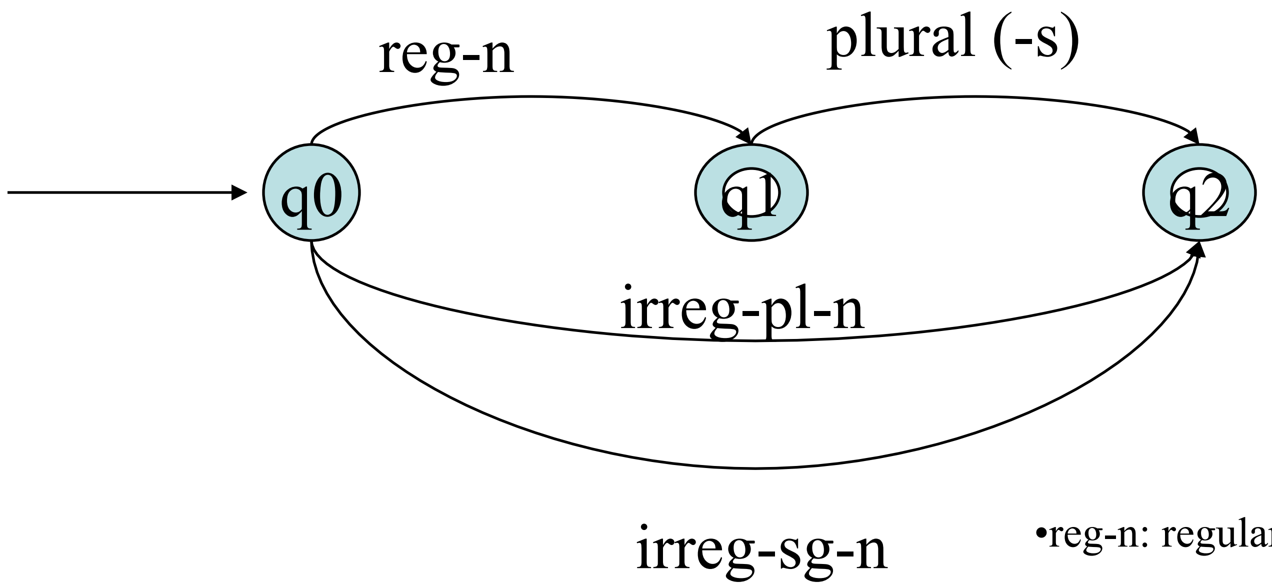
- **Many paths are possible…**
- **Start with compute**
  - **Computer -> computerize -> computerization**
  - **Computation -> computational**
  - **Computer -> computerize -> computerizable**
  - **Compute -> computee**

Templatic Morphology: Root Pattern Examples from Arabic

Word & Transliteration	Meaning	Word & Transliteration	Meaning
<naaâma> [نامَ]	He slept	*<naâ'imun>* [نائمٌ]	Sleeping
<yanaâmu> [ينامُ]	He sleeps	*<munawwamun>* [منوَّمٌ]	Under hypnotic
<nam> [نمْ]	Sleep	*<na'ûmun>* [نؤومٌ]	Late riser
<tanwçmun> [تنويمٌ]	Lulling to sleep	*<'anwamu>* [أنومَ]	More given to sleep
<manaâmun> [منامٌ]	Dream	*<nawwaâmun>* [نوّامٌ]	The most given to sleep
<nawmatun> [نومةٌ]	Of one sleep	*<manaâmun>* [منامٌ]	Dormitory
<nawwaâmatun> [نوامةٌ]	Sleeper	*<'an yanaâma>* [أن ينامَ]	That he sleeps
<nawmiyyatun> [نوميةٌ]	Pertaining to sleep	*<munawwamun>* [منوِّمٌ]	hypnotic

Morphotactic Models

- **English nominal inflection**



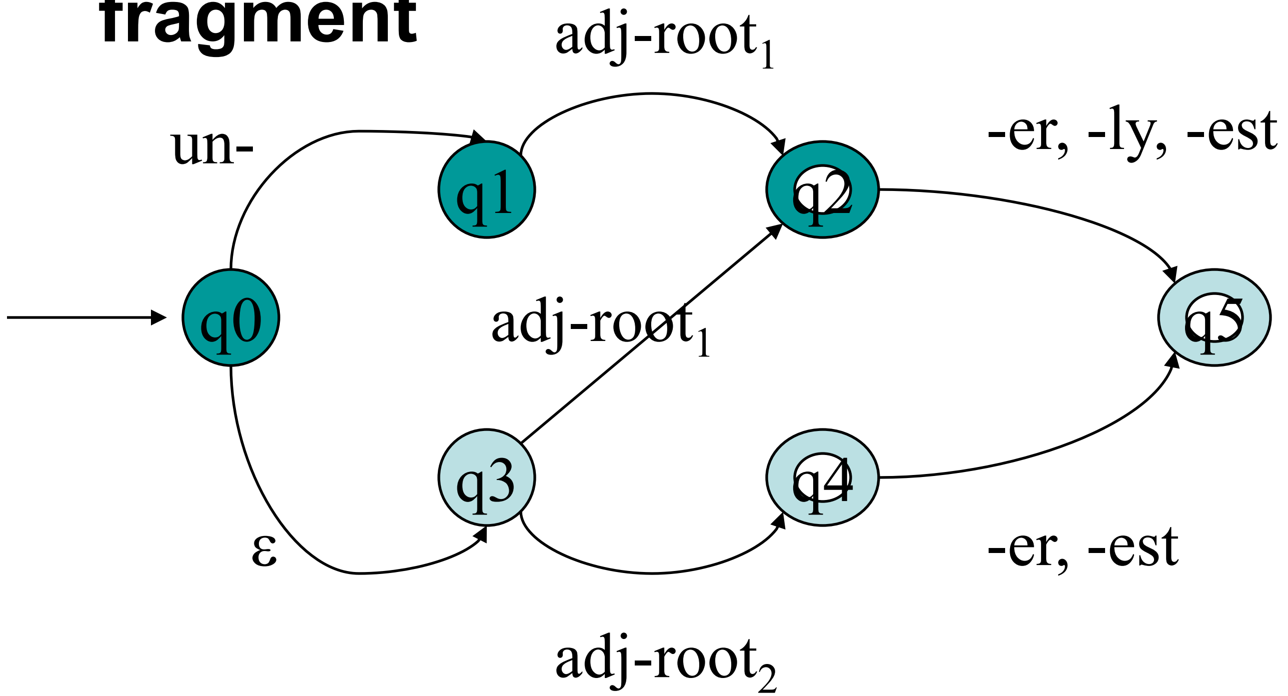
•reg-n: regular noun

•irreg-pl-n: irregular plural noun

•irreg-sg-n: irregular singular noun

•Inputs: cats, goose, geese

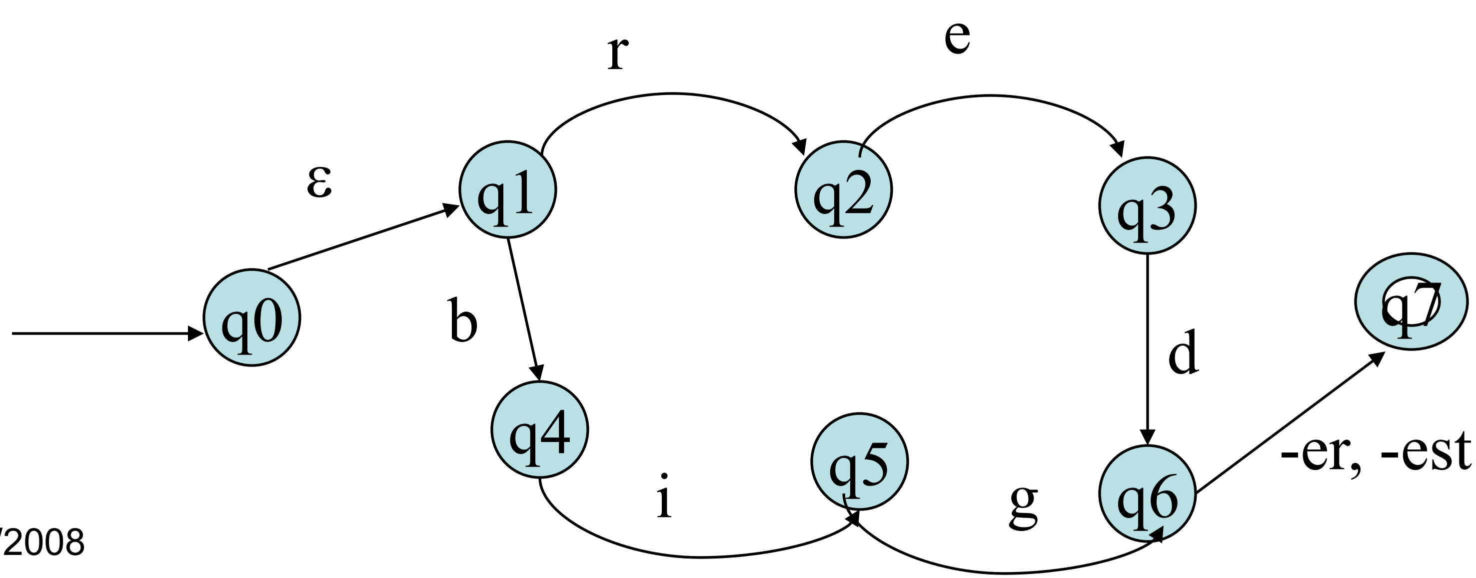
- **Derivational morphology: adjective fragment**



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

Using FSAs to Represent the Lexicon and Do Morphological Recognition

- **Lexicon: We can expand each non-terminal in our NFSA into each stem in its class (e.g. adj_root$_2$ = {big, red}) and expand each such stem to the letters it includes (e.g. red → r e d, big → b i g)**



Limitations

- **To cover all of English will require very large FSAs with consequent search problems**
  - **Adding new items to the lexicon means re-computing the FSA**
  - **Non-determinism**
- **FSAs can only tell us whether a word is in the language or not – what if we want to know more?**
  - **What is the stem?**
  - **What are the affixes?**
  - **We used this information to build our FSA: can we get it back?**

Parsing with Finite State Transducers

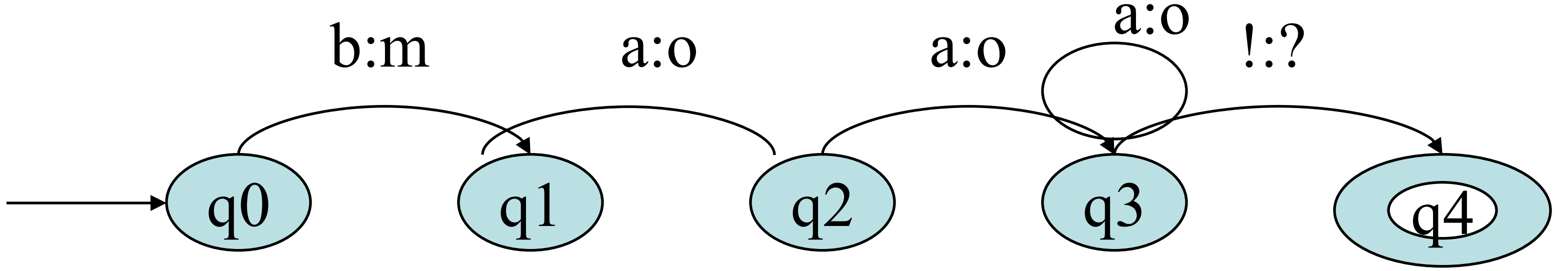
- **cats →cat +N +PL**
- **Kimmo Koskenniemi's two-level morphology**
  - **Words represented as correspondences between** lexical **level (the morphemes) and** surface **level (the orthographic word)**
  - **Morphological parsing :building** mappings **between the lexical and surface levels**

	c	a	t	+N	+PL	
	c	a	t	s		

Finite State Transducers

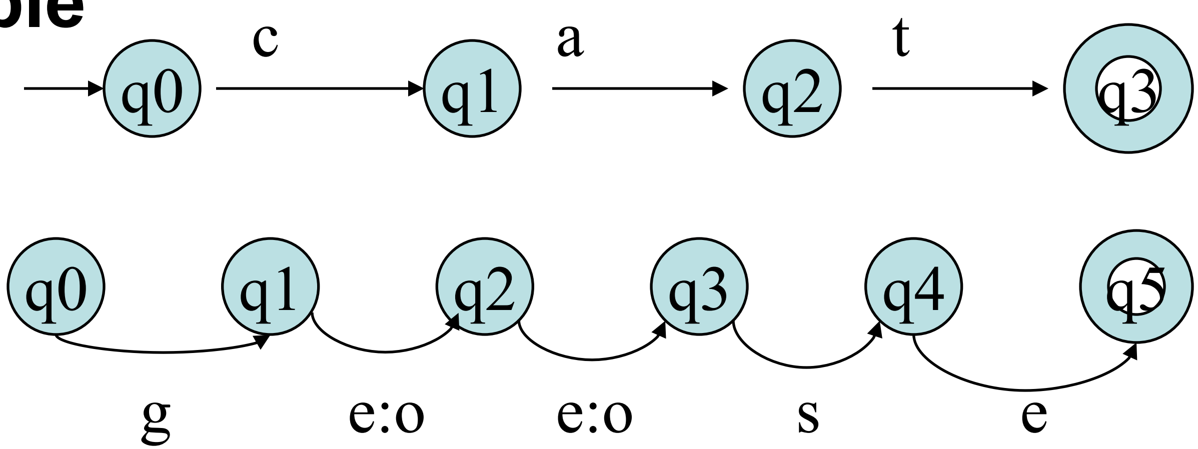
- **FSTs map between one set of symbols and another using an FSA whose alphabet Σ is composed of pairs of symbols from input and output alphabets**
- **In general, FSTs can be used for**
  - **Translator (Hello:مرحبا)**
  - **Parser/generator (Hello:How may I help you?)**
  - **To map between the lexical and surface levels of Kimmo's 2-level morphology**

- **FST is a 5-tuple consisting of**
  - **Q: set of states {q0,q1,q2,q3,q4}**
  - **Σ: an alphabet of complex symbols, each is an i/o pair such that i ∈ I (an input alphabet) and o ∈ O (an output alphabet) and Σ is in I x O**
  - **q0: a start state**
  - **F: a set of final states in Q {q4}**
  - **δ(q,i:o): a transition function mapping Q x Σ to Q**
  - Emphatic Sheep → Quizzical Cow



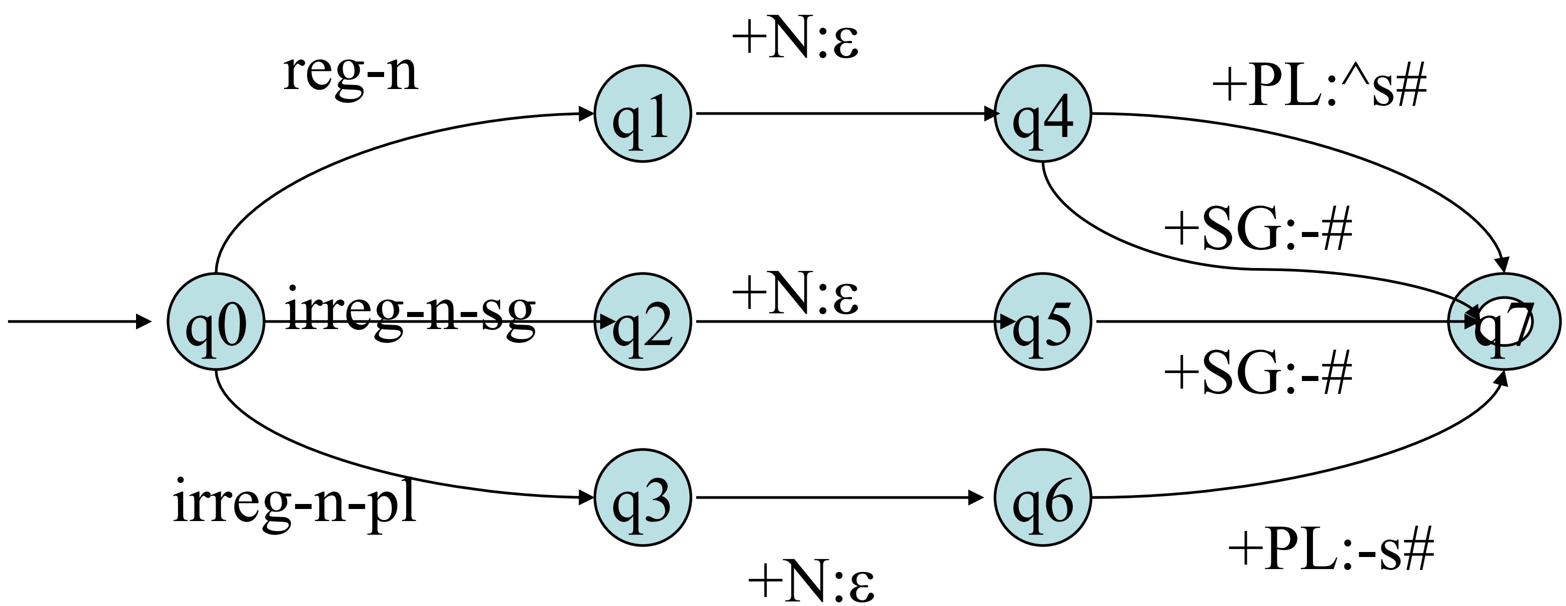
FST for a 2-level Lexicon

- **Example**



Reg-n	Irreg-pl-n	Irreg-sg-n
c a t	g o:e o:e s e	g o o s e

FST for English Nominal Inflection



Combining (cascade or composition) this FSA with FSAs for each noun type replaces e.g. reg-n with every regular noun representation in the lexicon

Orthographic Rules and FSTs

- **Define additional FSTs to implement rules such as consonant doubling (beg → begging), 'e' deletion (make → making), 'e' insertion (watch → watches), etc.**

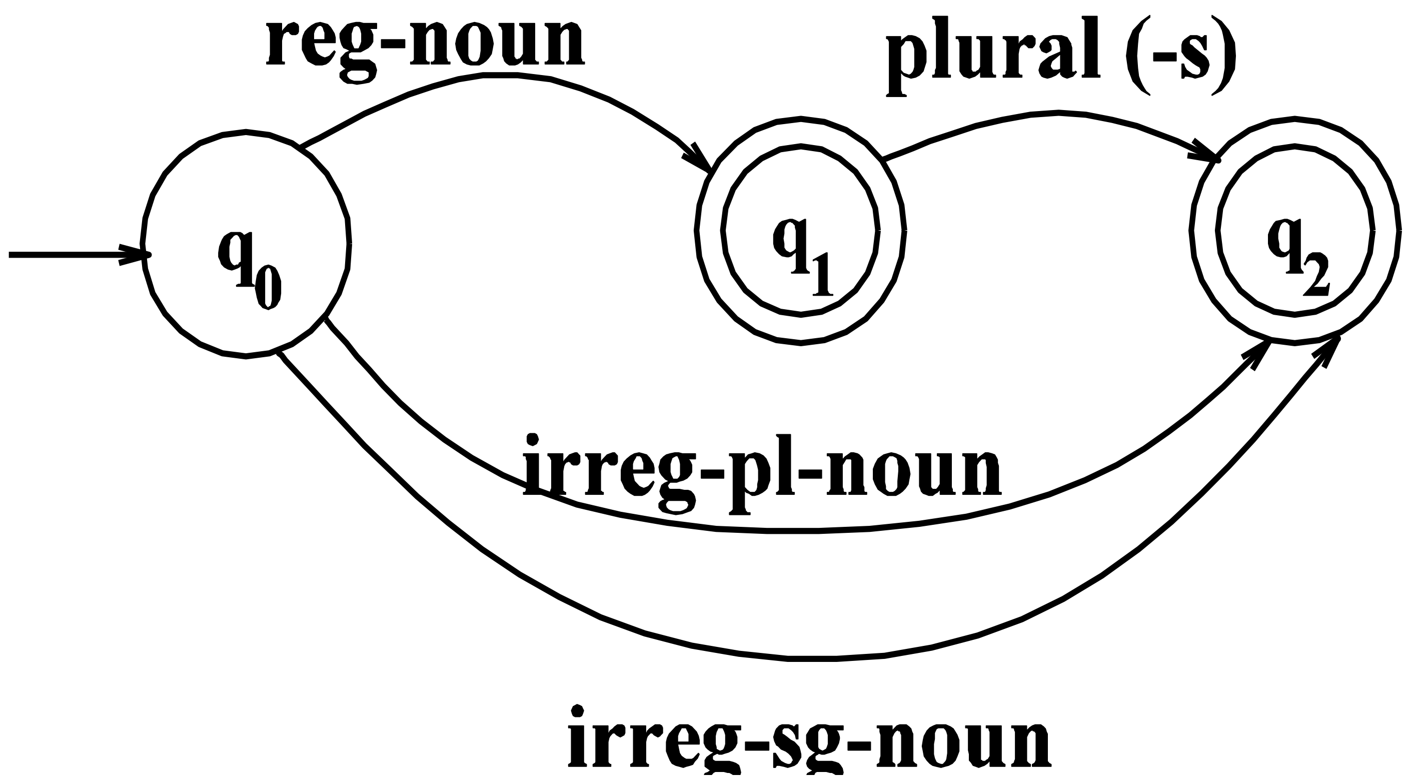
Lexical	**f**	**o**	**x**	**+N**	**+PL**	
Intermediate	**f**	**o**	**x**	**^**	**s**	**#**
Surface	**f**	**o**	**x**	**e**	**s**	

- **Note: These FSTs can be used for generation as well as recognition by simply exchanging the input and output alphabets (e.g. ^s#:+PL)**

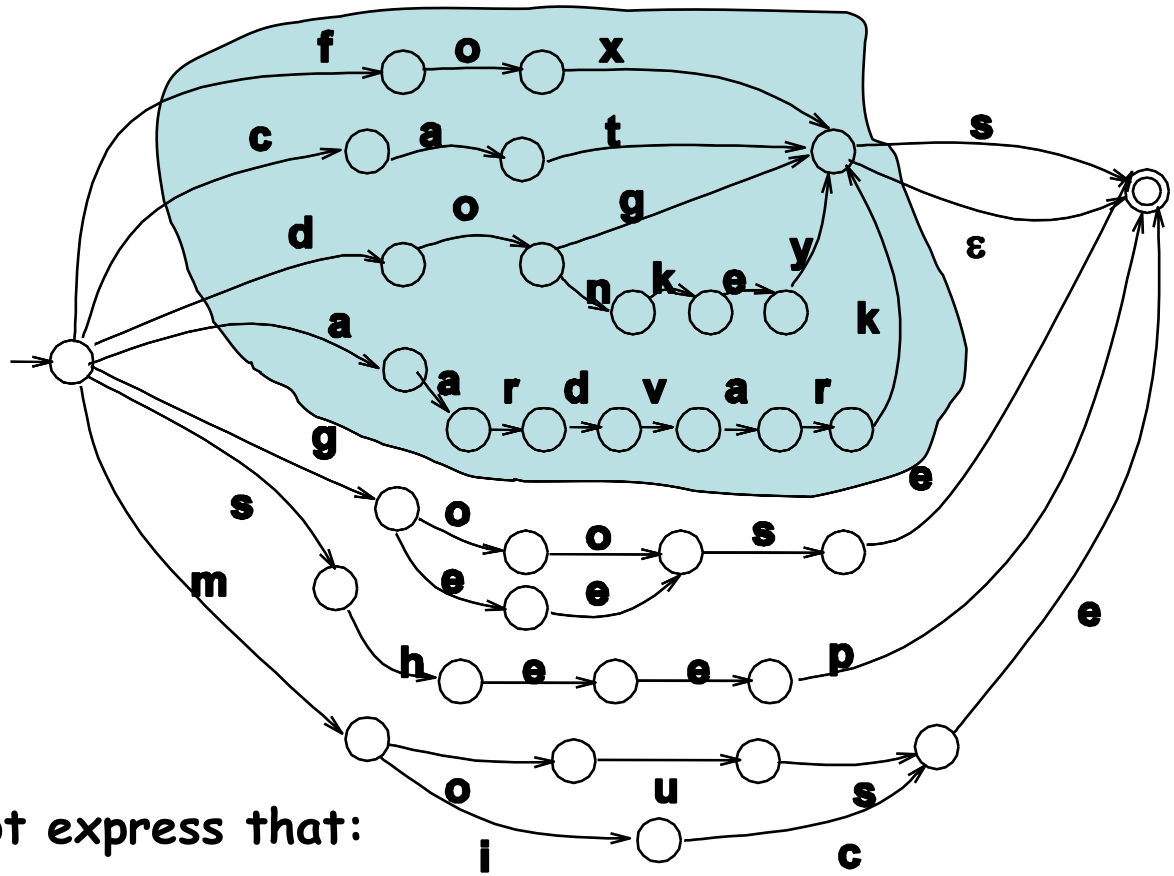
FSAs and the Lexicon

- **First we'll capture the *morphotactics***
  - **The rules governing the ordering of affixes in a language.**
- **Then we'll add in the actual *stems***

Simple Rules



Adding the Words

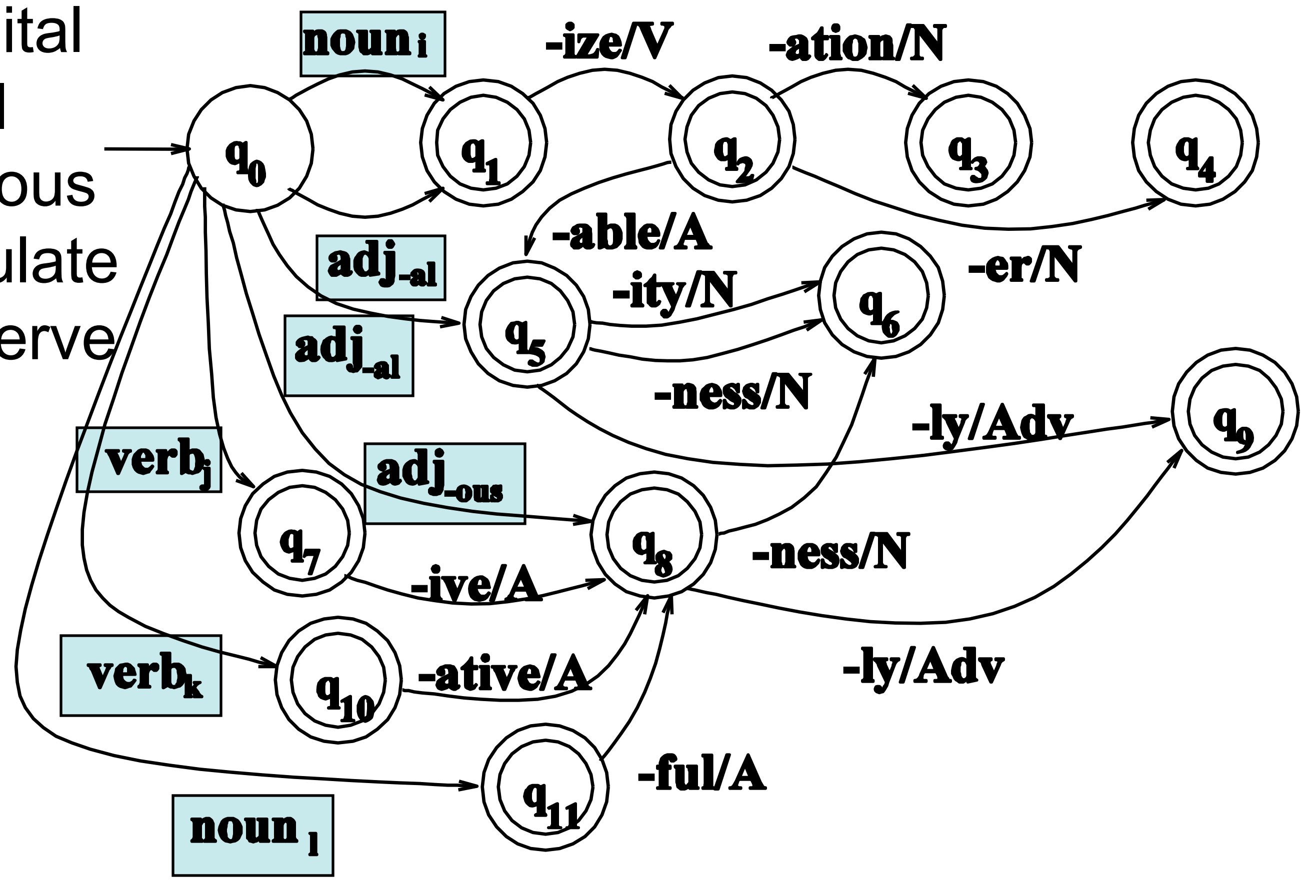


But it does not express that:

- Reg nouns ending in –s, -z, -sh, -ch, -x -> es (kiss, waltz, bush, rich, box)
- Reg nouns ending –y preceded by a consonant change the –y to -i

Derivational Rules

[noun$_i$] eg. hospital
[adj$_{al}$] eg. formal
[adj$_{ous}$] eg. arduous
[verb$_j$] eg. speculate
[verb$_k$] eg. conserve



Parsing/Generation vs. Recognition

- **Recognition is usually not quite what we need.**
  - **Usually if we find some string in the language we need to find the structure in it (parsing)**
  - **Or we have some structure and we want to produce a surface form (production/ generation)**

In other words

- **Given a word we need to find: the stem and its class and properties (parsing)**
- **Or we have a stem and its class and properties and we want to produce the word (production/generation)**
- **Example (parsing)**
  - **From "`cats`" to "`cat +N +PL`"**
  - **From "`lies`" to ……**

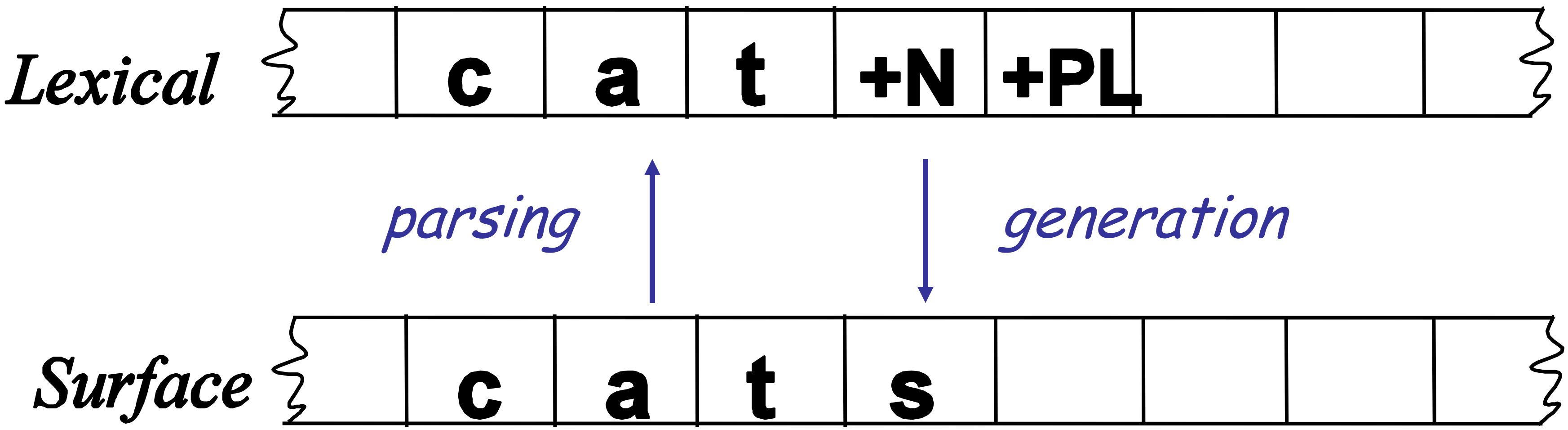
Applications

- **The kind of parsing we're talking about is normally called morphological analysis**
- **It can either be**
  - **An important stand-alone component of an application (spelling correction, information retrieval)**
  - **Or simply a link in a chain of processing**

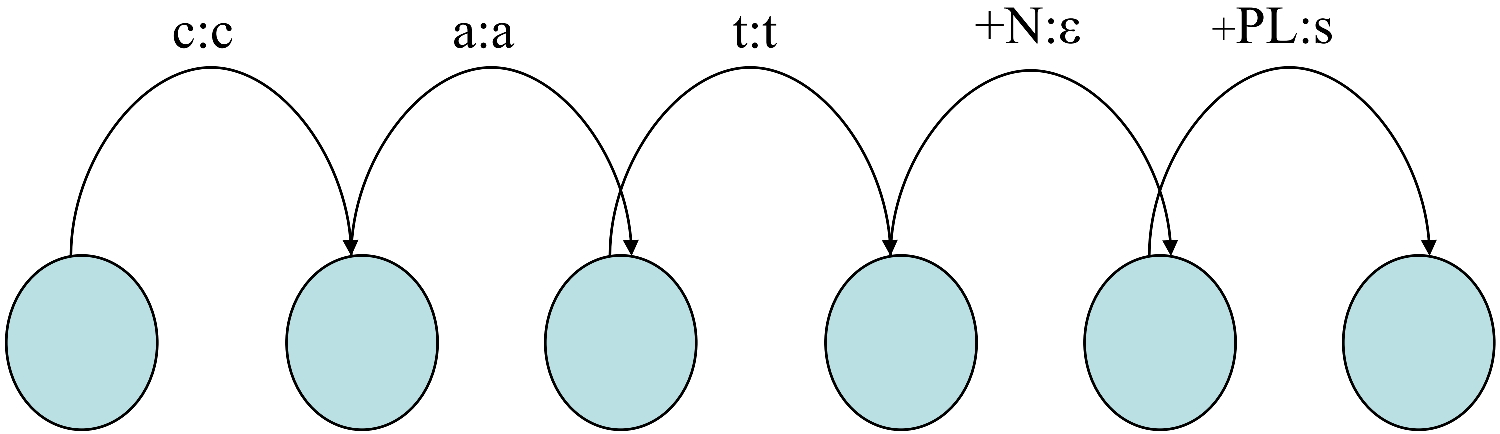
Finite State Transducers

- **The simple story**
  - **Add another tape**
  - **Add extra symbols to the transitions**
  - **On one tape we read "`cats`", on the other we write "`cat +N +PL`", or the other way around.**

FSTs



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**

Typical Uses

- **Typically, we'll read from one tape using the first symbol on the machine transitions (just as in a simple FSA).**
- **And we'll write to the second tape using the other symbols on the transitions.**

Ambiguity

- **Recall that in non-deterministic recognition multiple paths through a machine may lead to an accept state.**
  - **Didn't matter which path was actually traversed**
- **In FSTs the path to an accept state does matter since different paths represent different parses and different outputs will result**

Ambiguity

- **What's the right parse for**
  - **Unionizable**
  - **Union-ize-able**
  - **Un-ion-ize-able**
- **Each represents a valid path through the derivational morphology machine.**

Ambiguity

- **There are a number of ways to deal with this problem**
  - **Simply take the first output found**
  - **Find all the possible outputs (all paths) and return them all (without choosing)**
  - **Bias the search so that only one or a few likely paths are explored**

More Details

- **Its not always as easy as**
  - `“cat +N +PL” <-> “cats”`
- **There are geese, mice and oxen**
- **There are also spelling/ pronunciation changes that go along with inflectional changes**

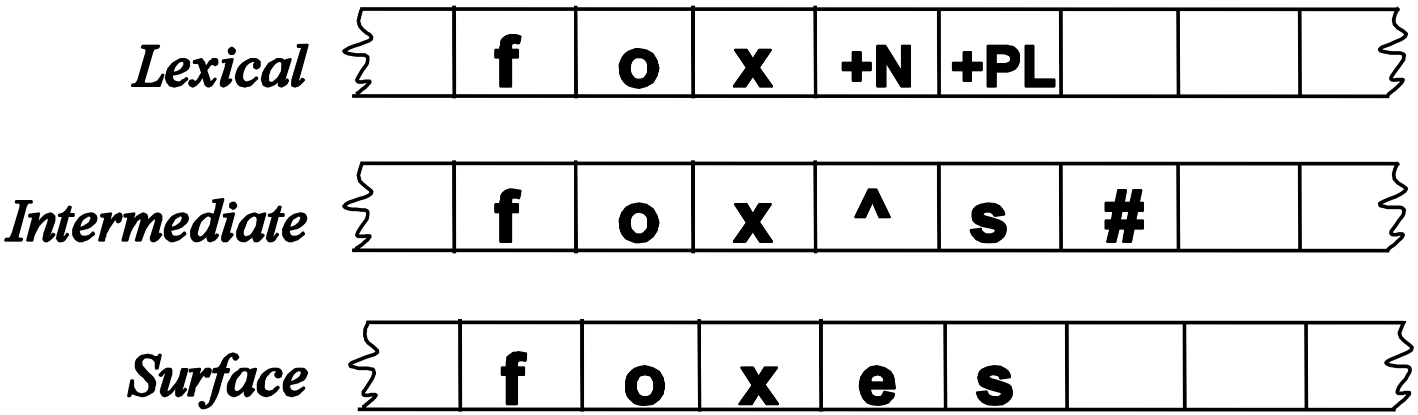
Multi-Tape Machines

- **To deal with this we can simply add more tapes and use the output of one tape machine as the input to the next**
- **So to handle irregular spelling changes we'll add intermediate tapes with intermediate symbols**

Spelling Rules and FSTs

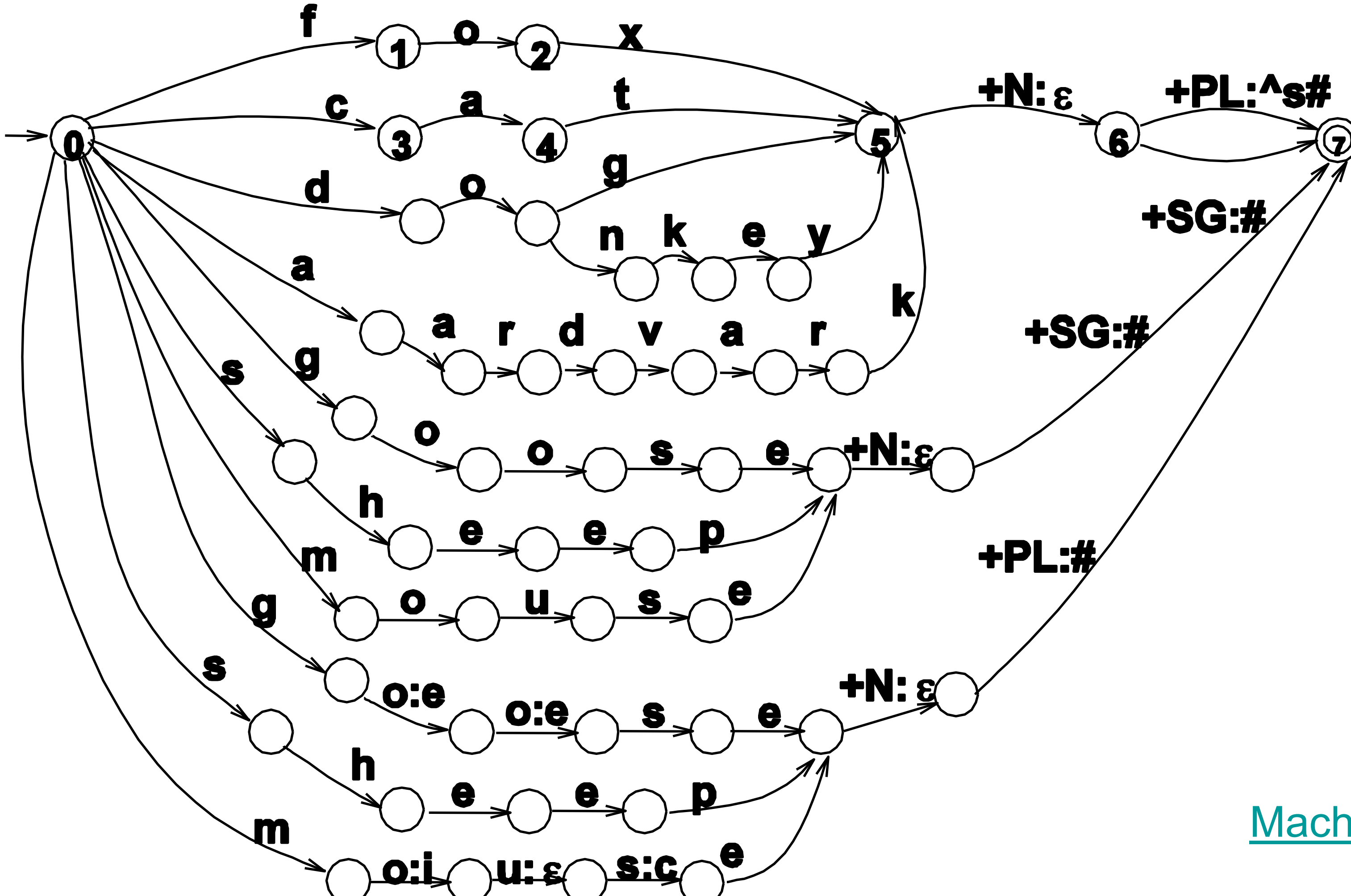
Name	Description of Rule	Example
Consonant doubling	1-letter consonant doubled before *-ing/-ed*	beg/begging
E deletion	Silent e dropped before -ing and –ed	make/making
E insertion	e added after *–s, -z, -x, -ch, -sh* before *-s*	watch/watches
Y replacement	*-y* changes to *–ie* before *-s*, and to *-i* before *-ed*	try/tries
K insertion	verbs ending with *vowel + -c* add *-k*	panic/panicked

Multi-Level Tape Machines



- **We use one machine to transducer between the lexical and the intermediate level, and another to handle the spelling changes to the surface tape**

Lexical to Intermediate Level

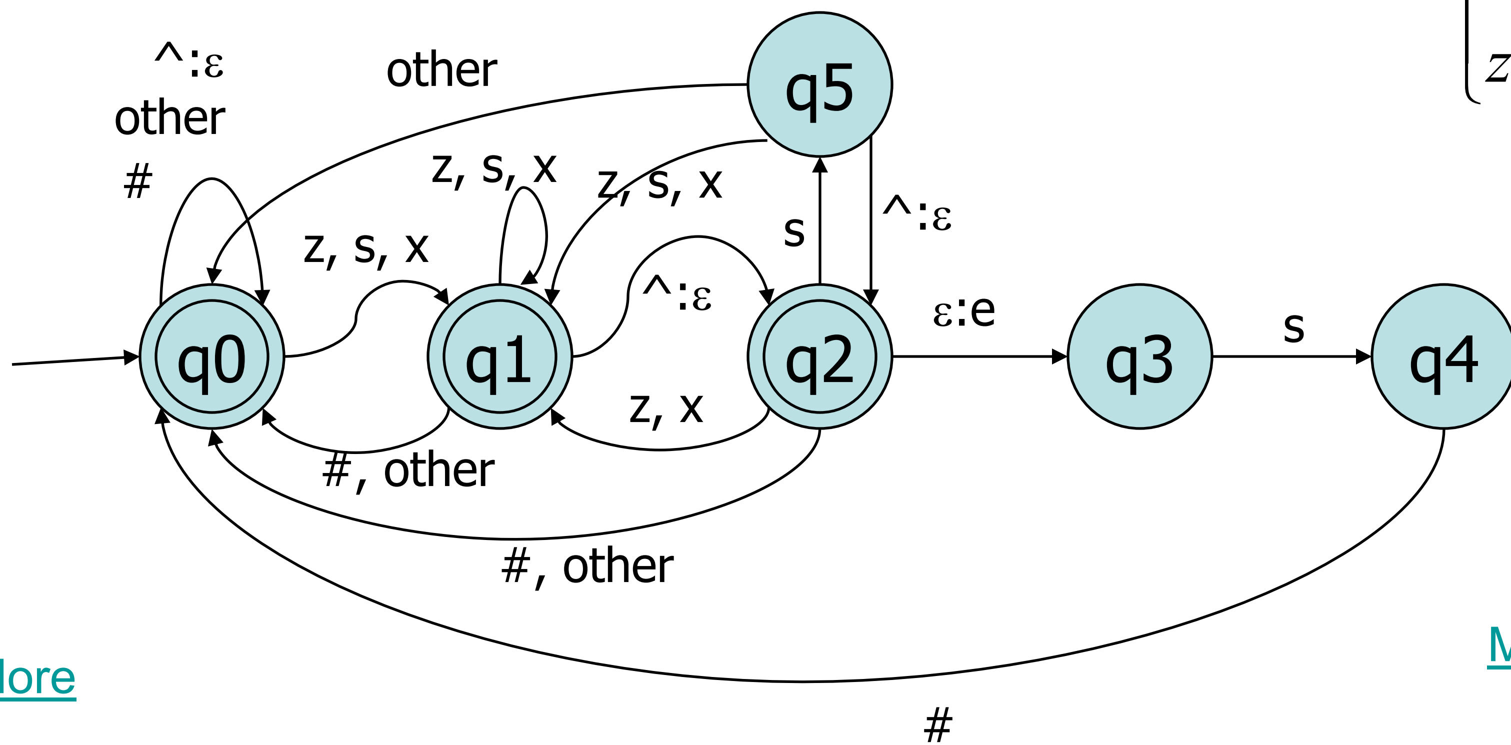


Machine

FST for the E-insertion Rule: Intermediate to Surface

- **The add an “e” rule as in**
  `fox^s# <-> foxes`

$$\varepsilon \rightarrow e \,/\, \left\{ \begin{matrix} x \\ s \\ z \end{matrix} \right\} \hat{}\ \underline{\quad}\ s\#$$



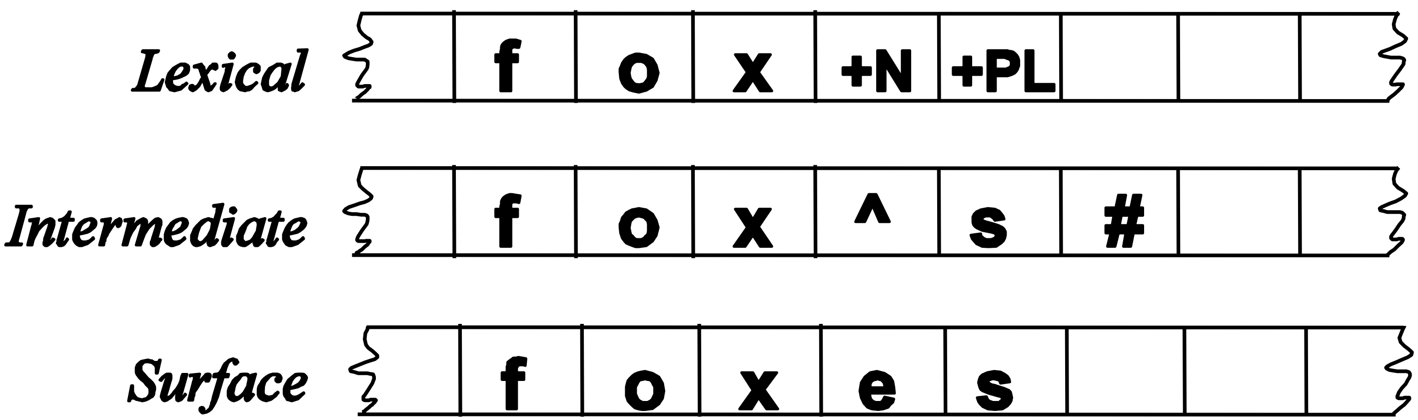
More

Machine

Note

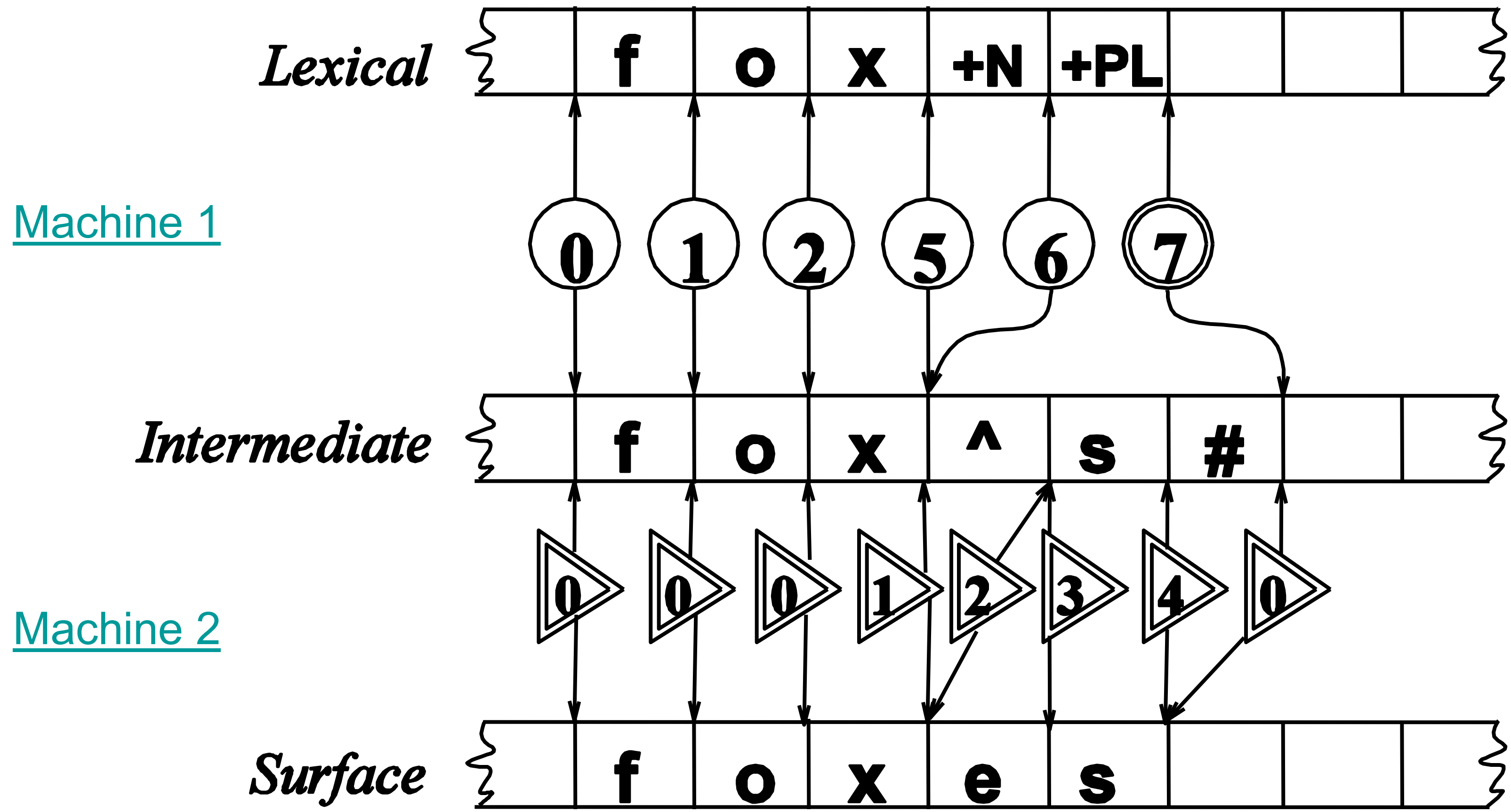
- **A key feature of this machine is that it doesn’t do anything to inputs to which it doesn’t apply.**
- **Meaning that: they are written out unchanged to the output tape.**

English Spelling Changes



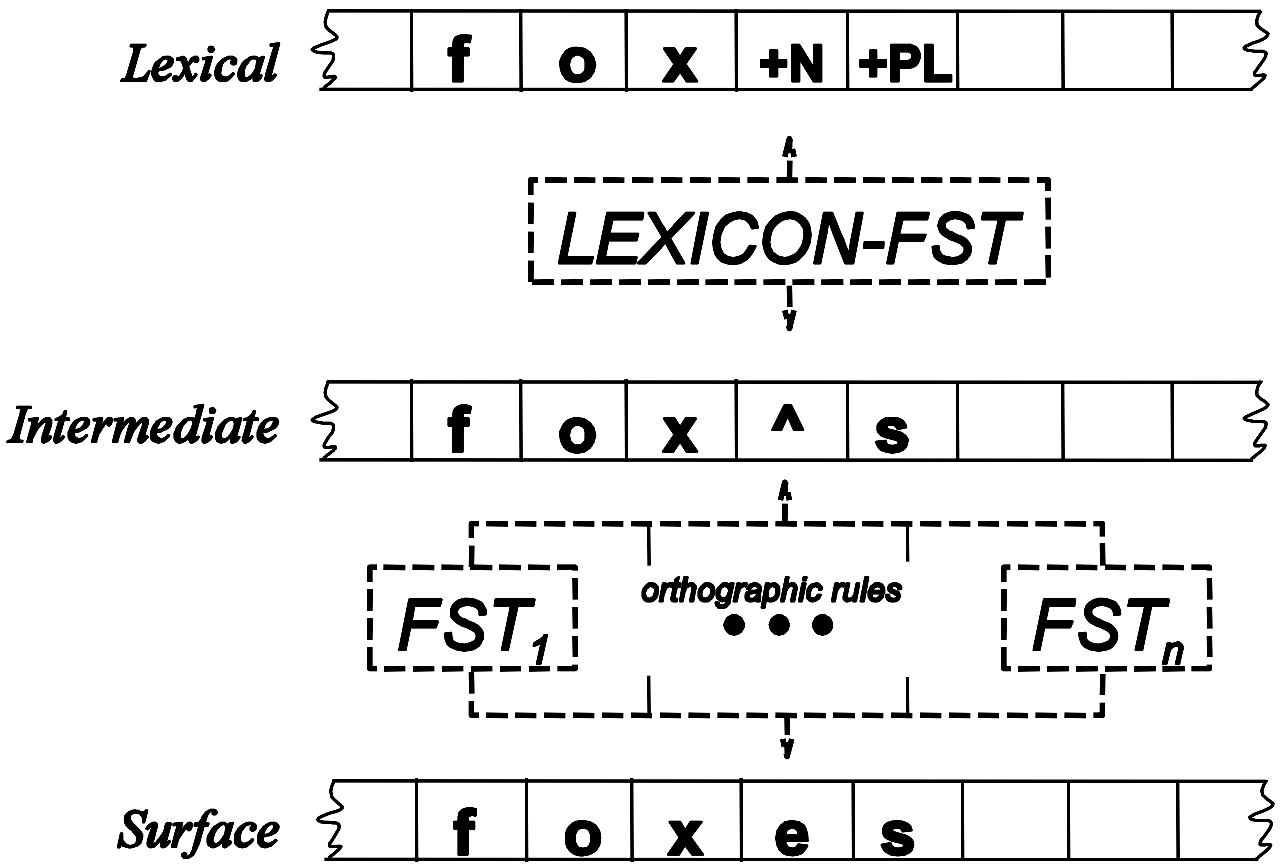
- **We use one machine to transduce between the lexical and the intermediate level, and another to handle the spelling changes to the surface tape**

Foxes

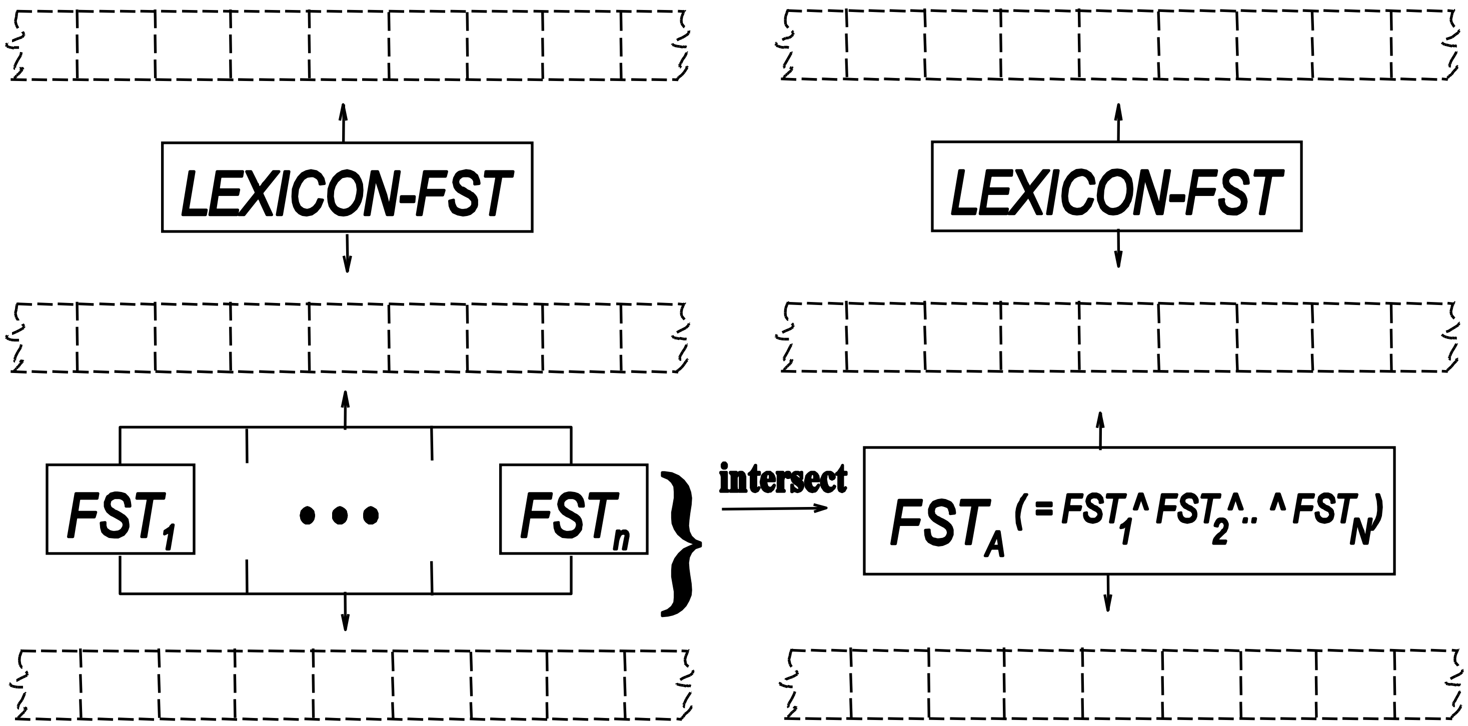


Lexical
f
o
x
+N
+PL
Machine 1
0
1
2
5
6
7
Intermediate
f
o
x
^
s
#
Machine 2
0
0
0
1
2
3
4
0
Surface
f
o
x
e
s

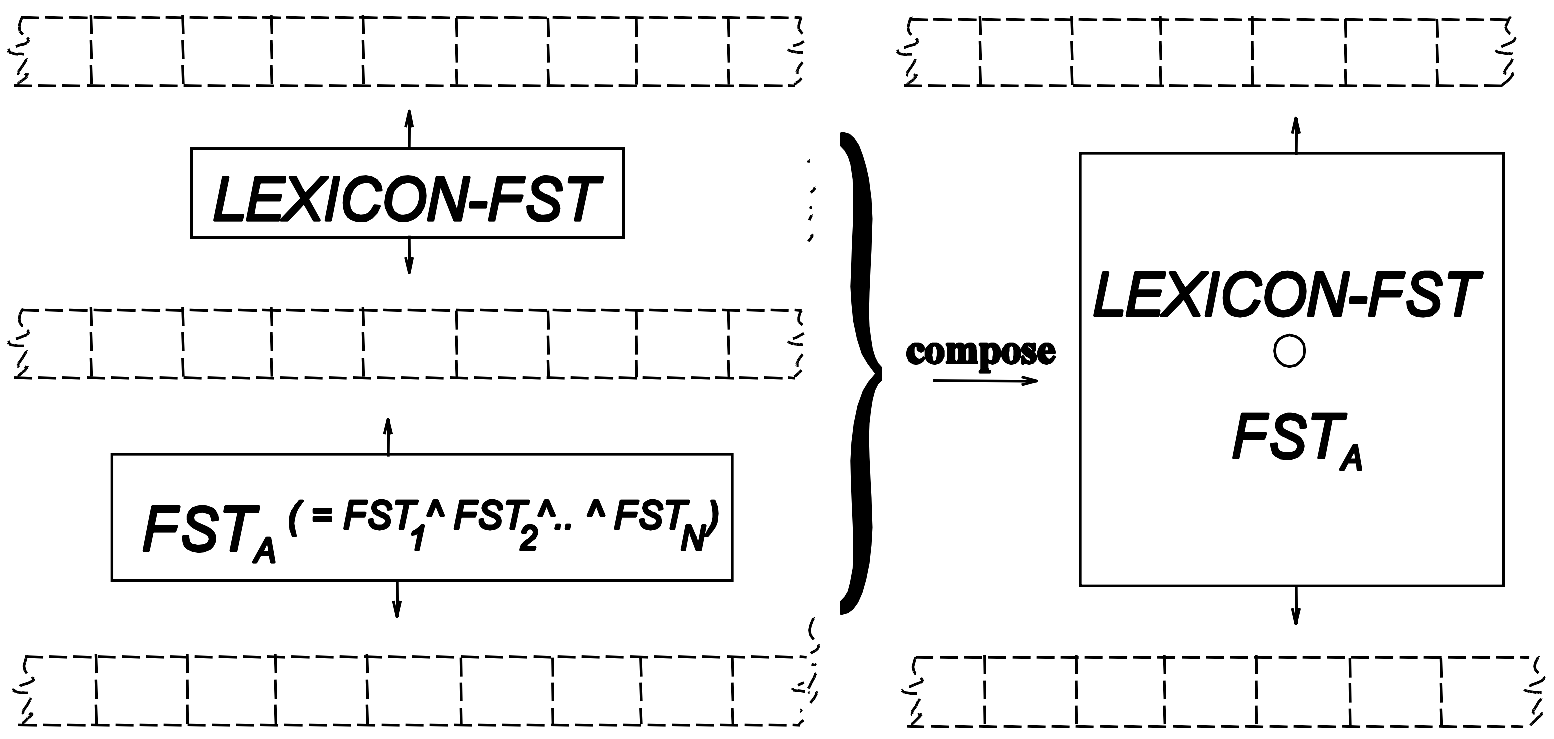
Overall Plan



Final Scheme: Part 1



Final Scheme: Part 2



Stemming vs Morphology

- **Sometimes you just need to know the stem of a word and you don't care about the structure.**
- **In fact you may not even care if you get the right stem, as long as you get a consistent string.**
- **This is stemming… it most often shows up in IR (Information Retrieval) applications**

Stemming in IR

- **Run a stemmer on the documents to be indexed**
- **Run a stemmer on users queries**
- **Match**
  - **This is basically a form of hashing**

Porter Stemmer

- **No lexicon needed**
- **Basically a set of staged sets of rewrite rules that strip suffixes**
- **Handles both inflectional and derivational suffixes**
- **Doesn’t guarantee that the resulting stem is really a stem**
- **Lack of guarantee doesn’t matter for IR**

Porter Example

- **Computerization**
  - **ization -> -ize computerize**
  - **ize -> ε computer**
- **Other Rules**
  - **ing -> ε (motoring -> motor)**
  - **ational -> ate (relational -> relate)**
- **Practice: See Poter's Stemmer at Appendix B and suggest some rules for A KFUPM Arabic Stemmer**

Porter Stemmer

- **The original exposition of the Porter stemmer did not describe it as a transducer but…**
  - **Each stage is separate transducer**
  - **The stages can be composed to get one big transducer**

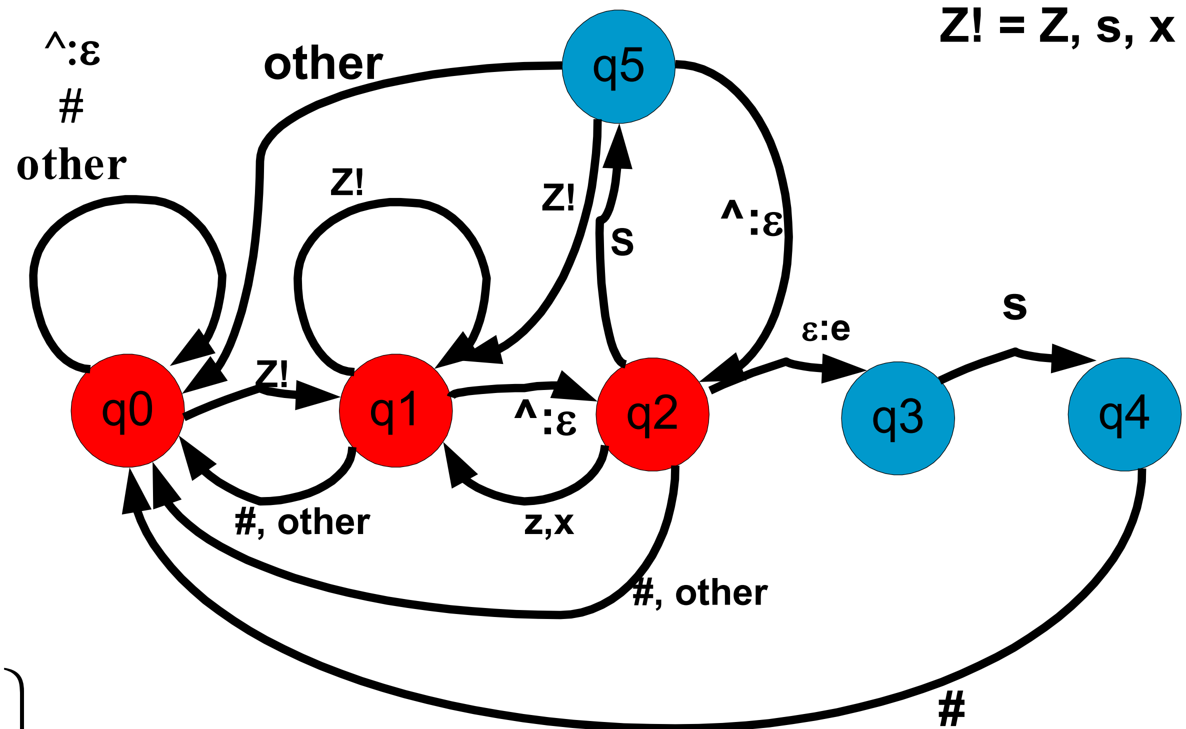
- **Hypotheses:**
  - Full listing hypothesis: **words listed**
  - Minimum redundancy hypothesis: **morphemes listed**
- **Experimental evidence:**
  - Priming **experiments (Does seeing/ hearing one word facilitate recognition of another?)**
  - **Regularly inflected forms prime stem but not derived forms**
  - **But *spoken* derived words can prime stems if they are semantically close (e.g. government/govern but not department/depart)**

Reminder: Quiz 1 Next class

- **Next time: Quiz**
  - **Ch 1!, 2, & 3 (Lecture presentations)**
  - **Do you need a sample quiz?**
    - **What is the difference between a sample and a template?**
    - **Let me think – It might appear at the WebCt site on late Saturday.**

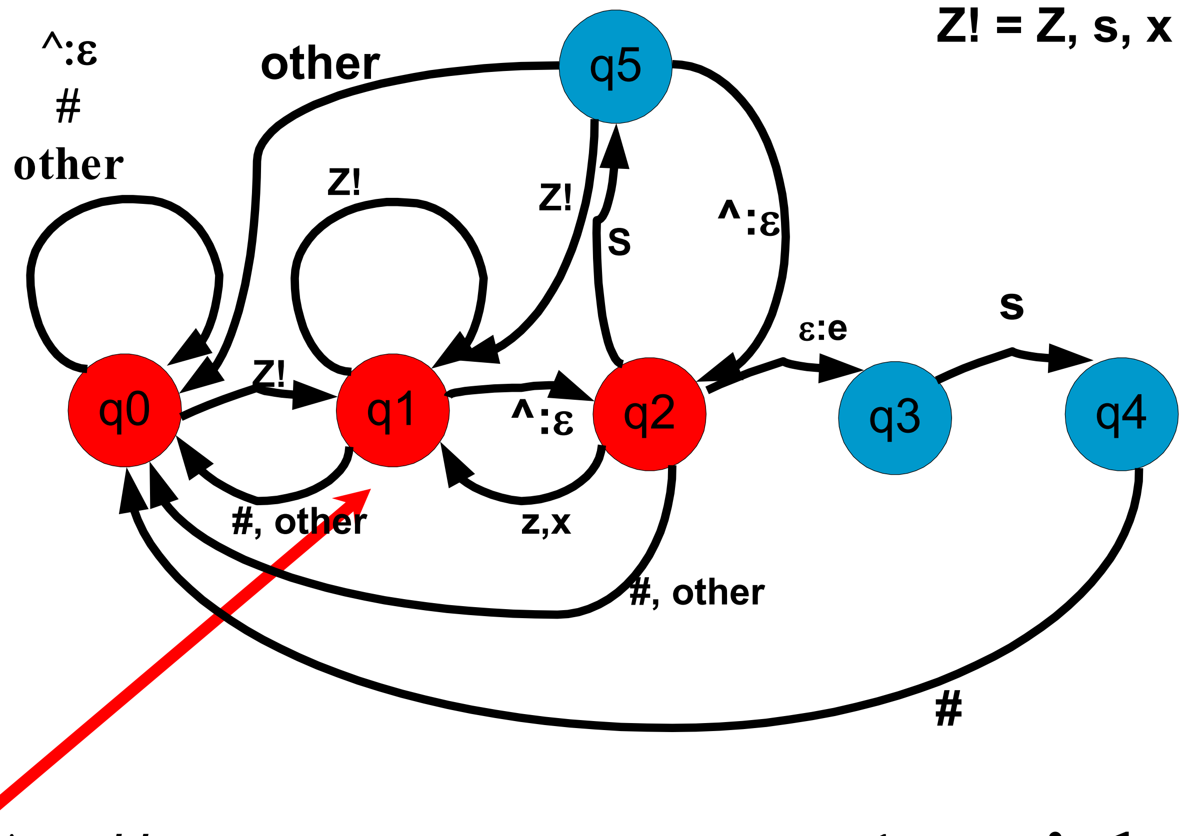
More Examples

Using FSTs for orthographic rules



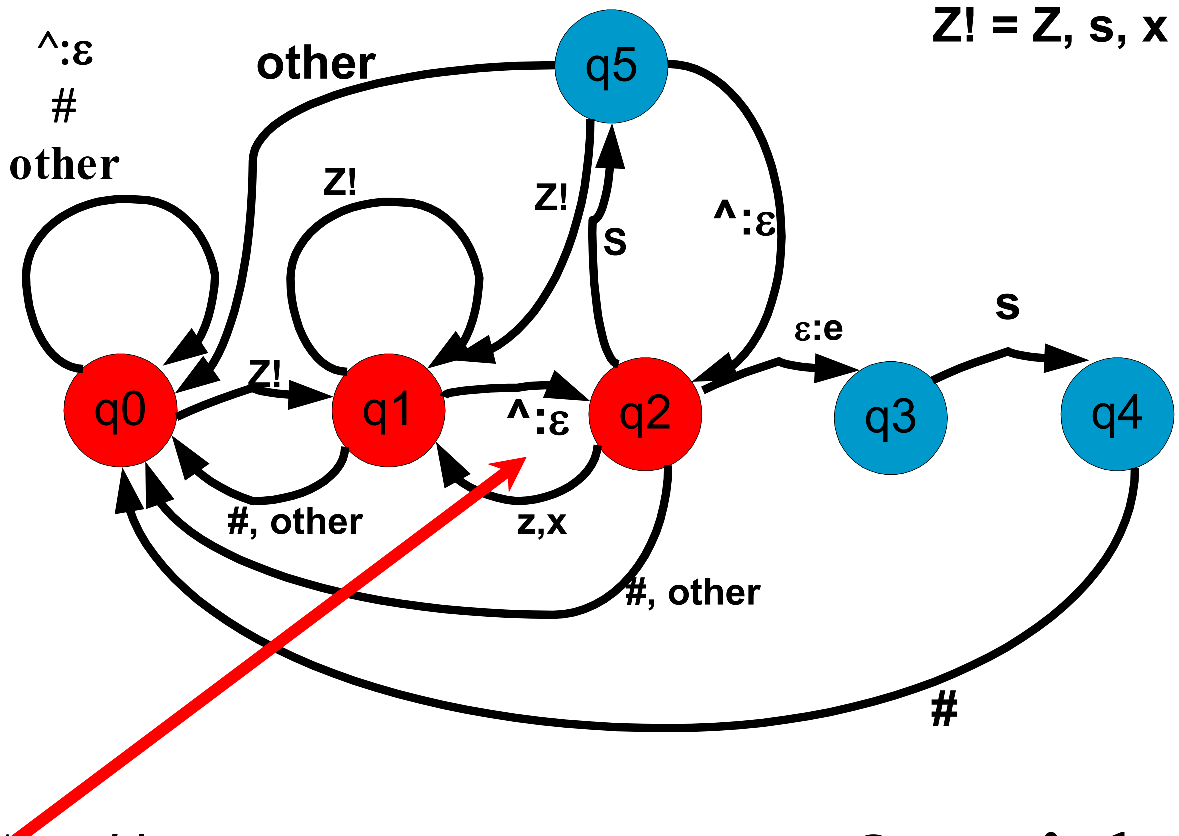
$$\varepsilon \rightarrow e / \left\{ \begin{matrix} x \\ s \\ z \end{matrix} \right\} \wedge \_\_ s\#$$

Using FSTs for orthographic rules



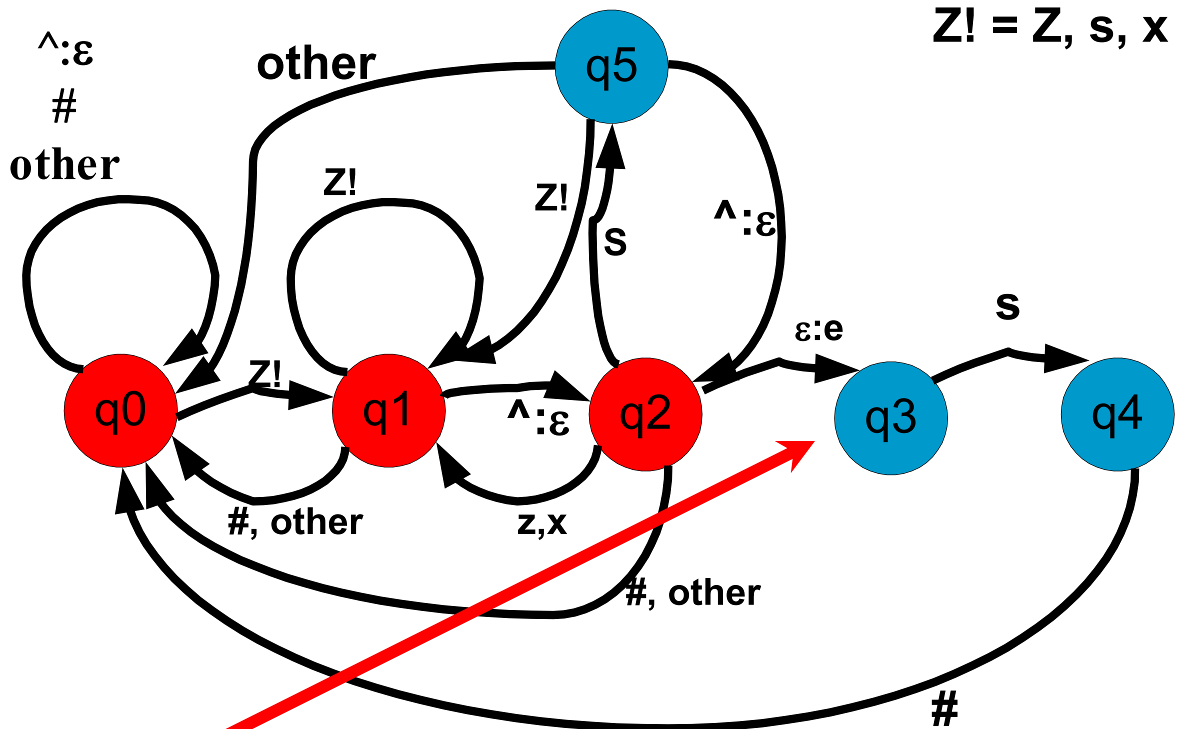
fox^s#…we get to q1 with ‘x’

Using FSTs for orthographic rules



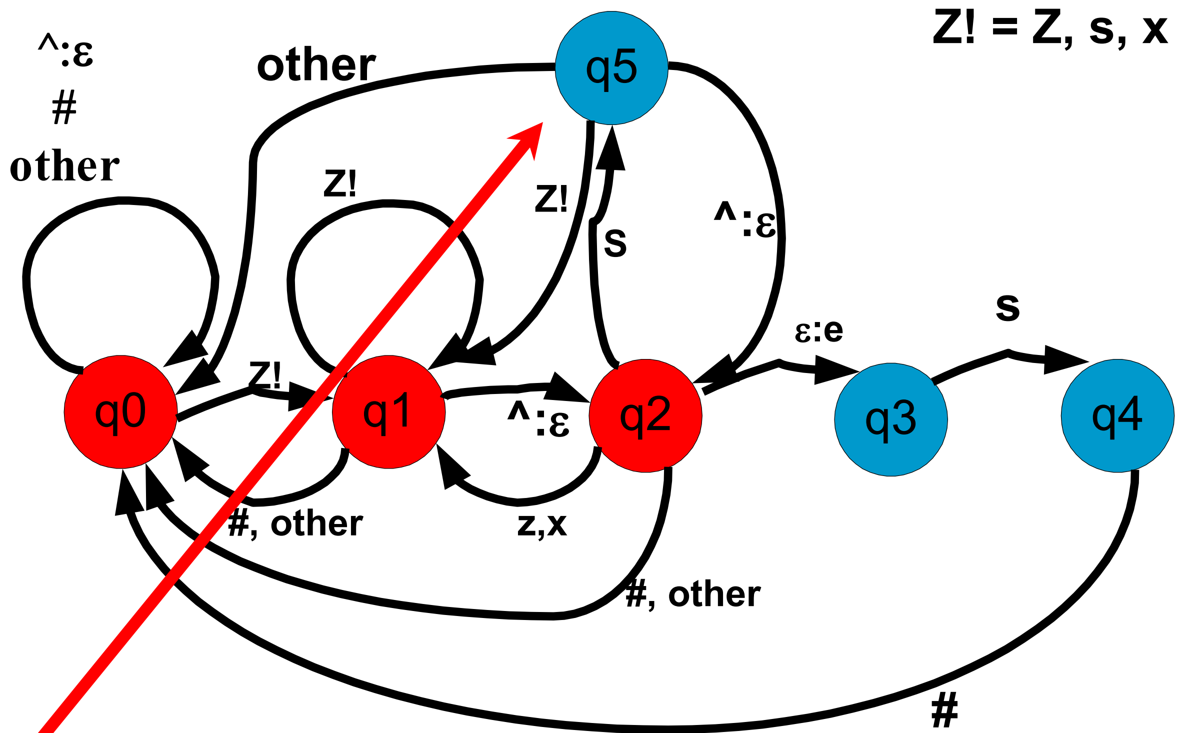
fox^s#…we get to q2 with '^'

Using FSTs for orthographic rules



fox^s#…we *can* get to q3 with 'NULL'

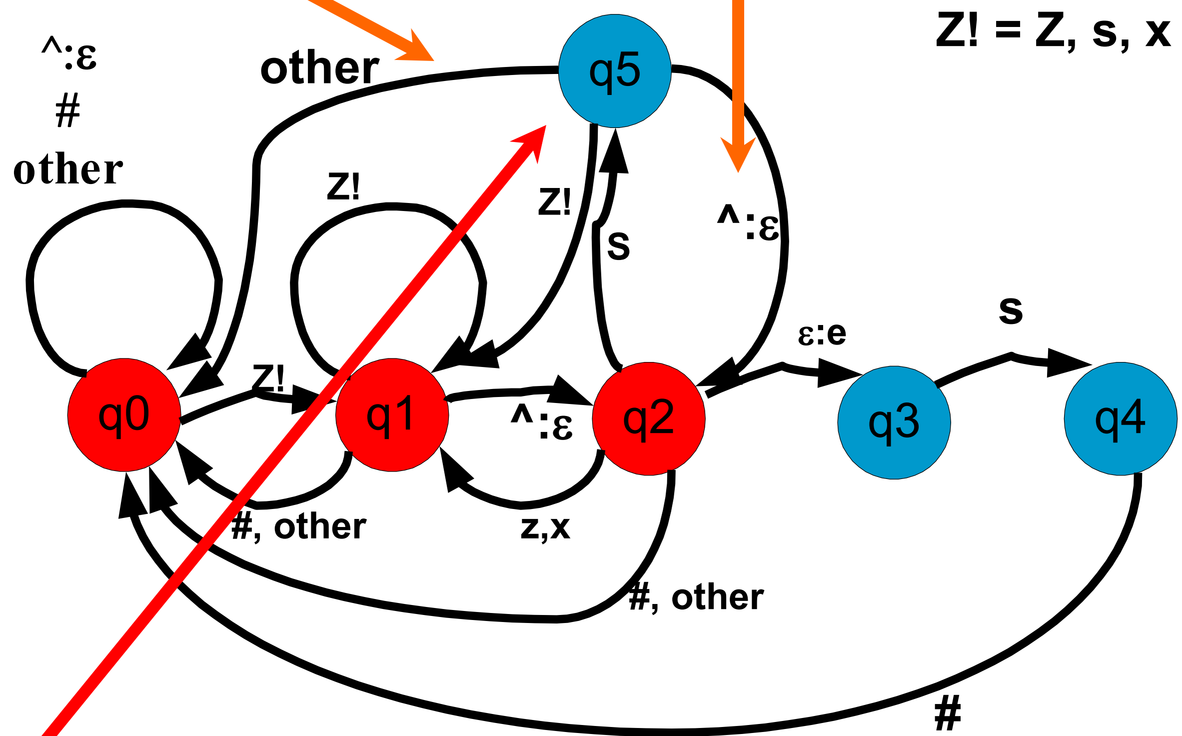
Using FSTs for orthographic rules



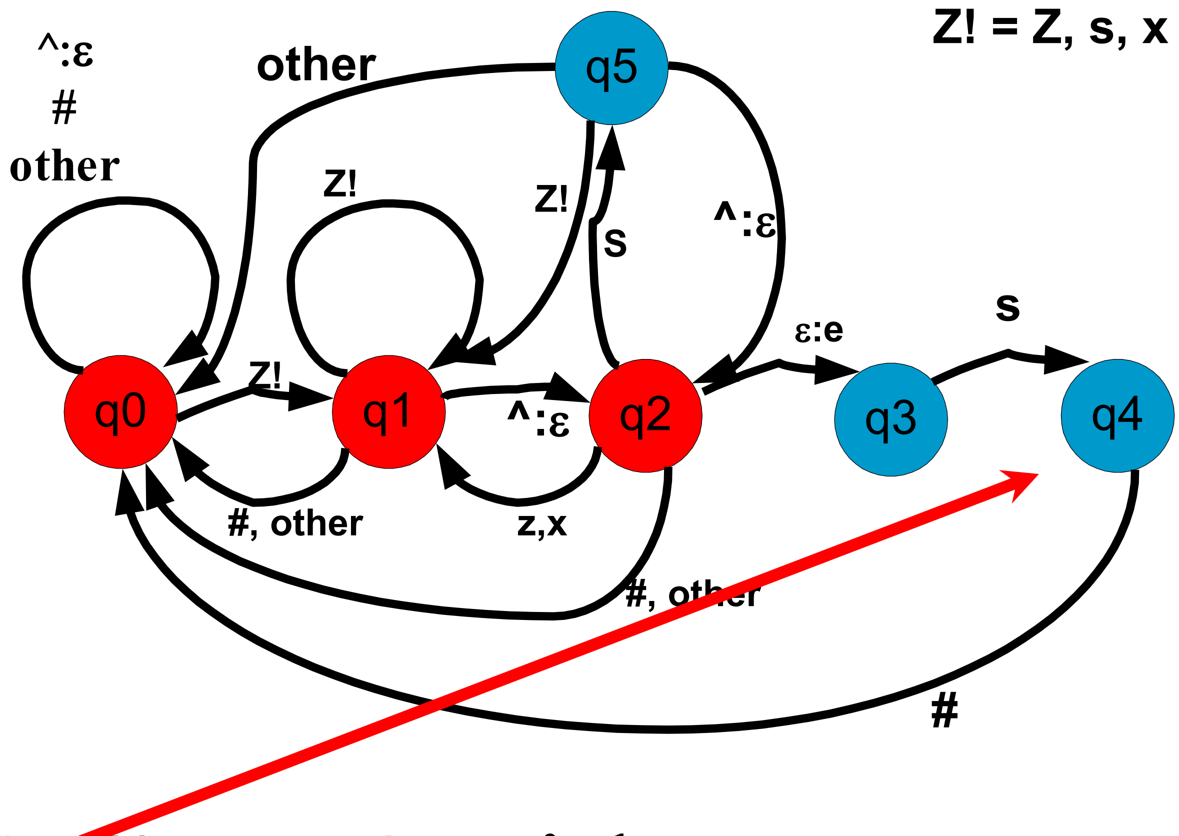
fox^s#…we ***also*** get to q5 with 's' but we don't want to!

So why is this transition there?

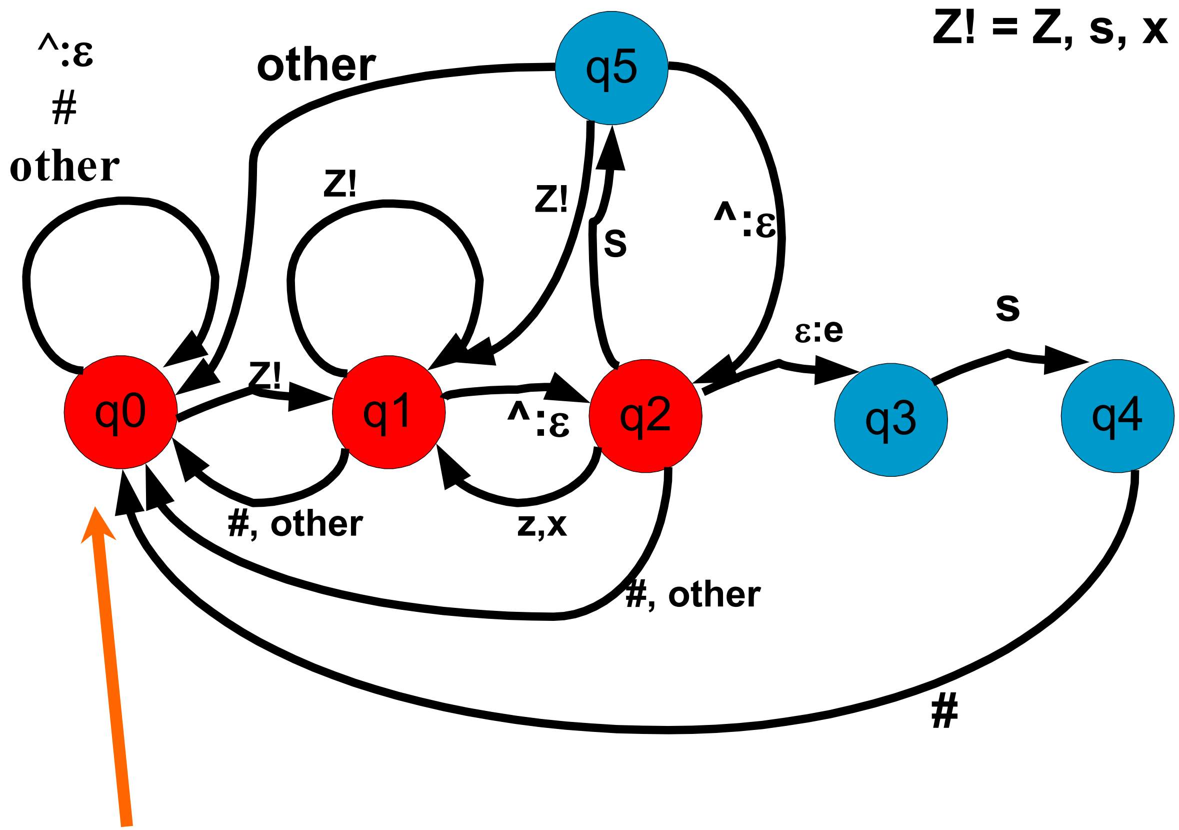
?friend^ship, ?fox^s^s (= foxes's)



fox^s#…we also get to q5 with 's' but we don't want to!



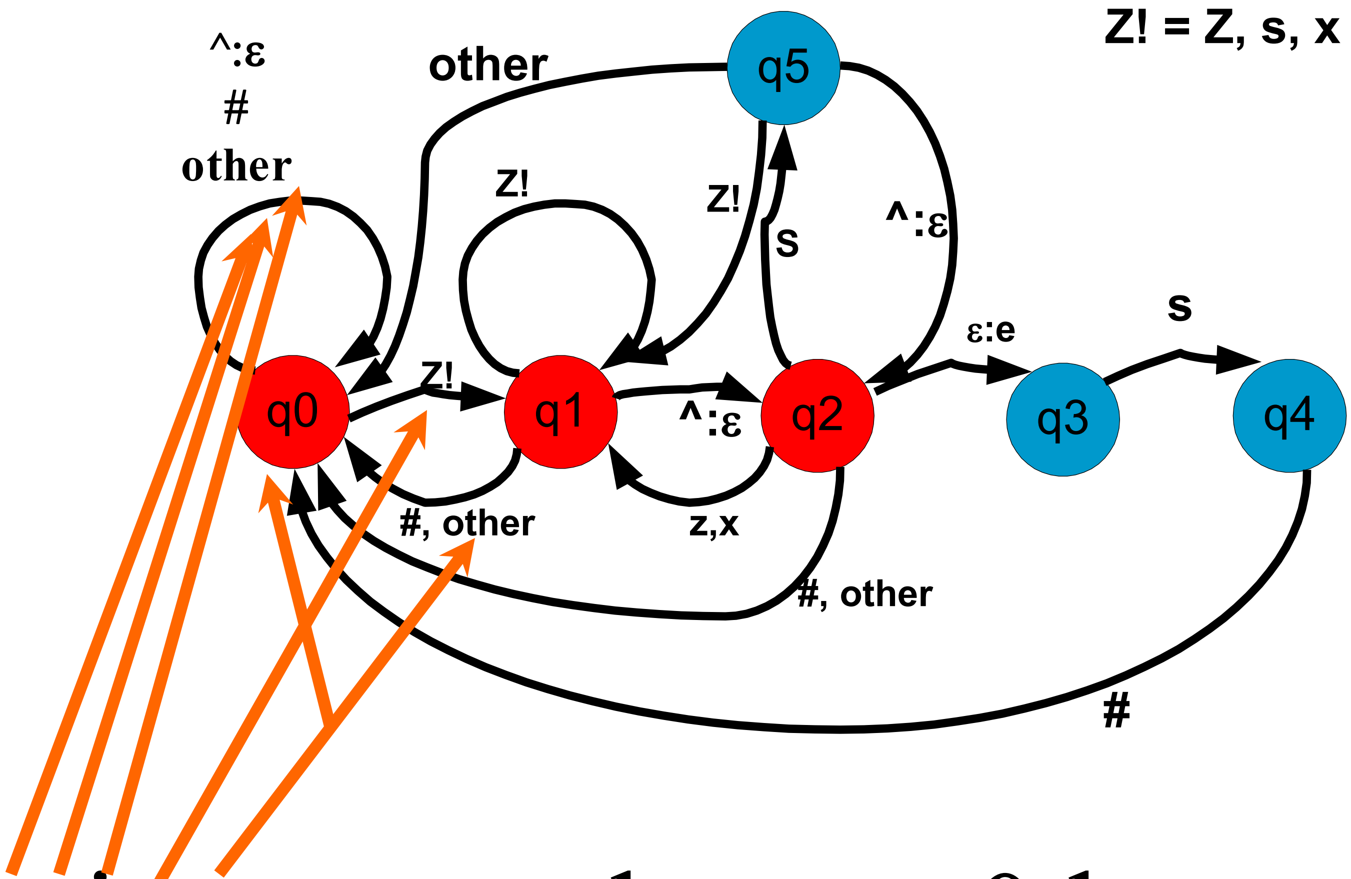
fox^s#…q4 with s



fox^s#…q0 with # (accepting state)

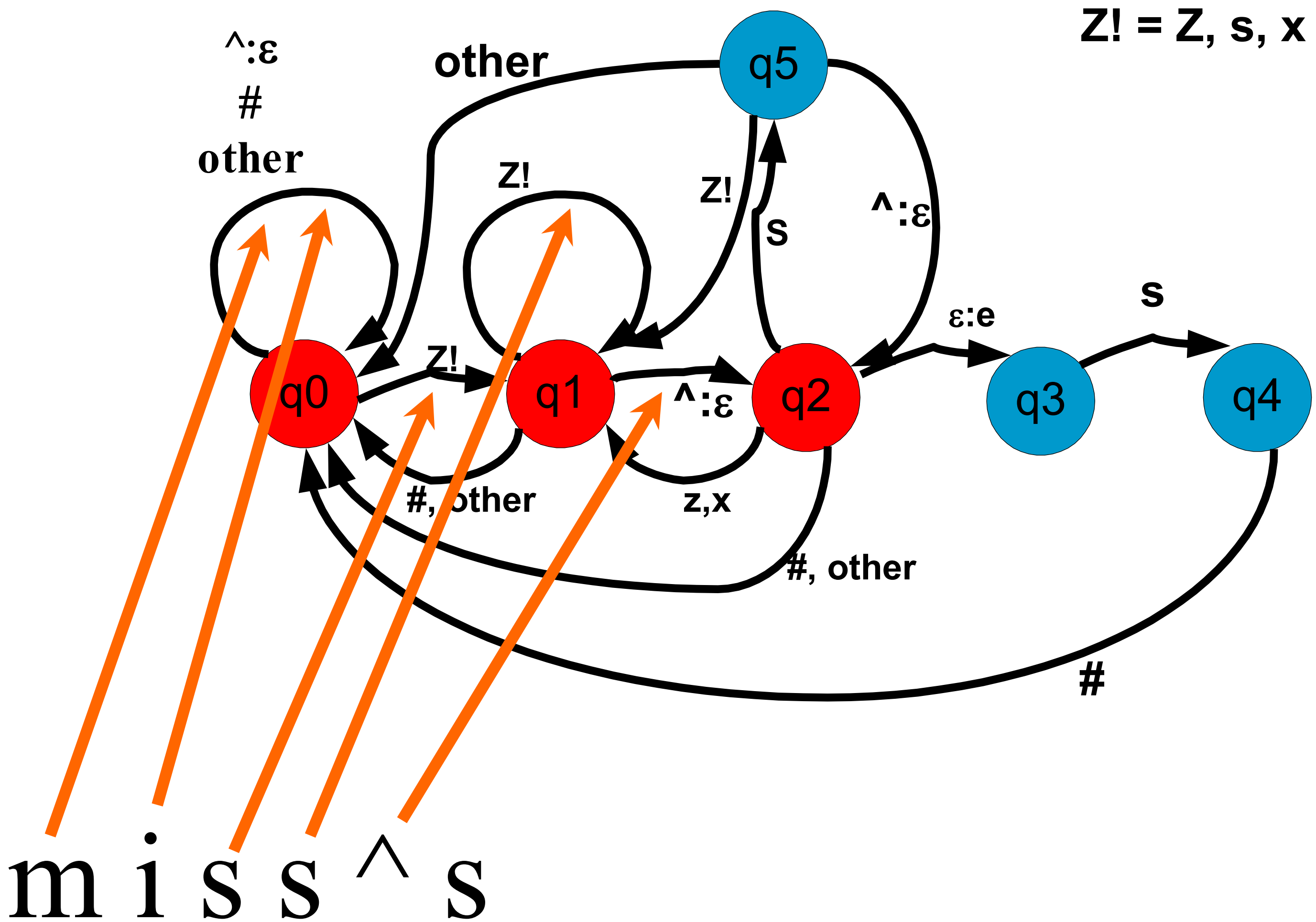
Back

Other transitions…



arizona: we leave q0 but return

Other transitions…



السلام عليكم ورحمة الله

سبحانك اللهم وبحمدك أشهد أن لا إله إلا أنت أستغفرك وأتوب اليك