Tutorial 2: Word-Level Processing

Q1. Consider the following word segmentation algorithm in the lecture notes:

> Given a lexicon of Chinese, and a string
>
> 1) Start a pointer at the beginning of the string
>
> 2) Find the longest word in dictionary that matches the string starting at pointer
>
> 3) Move the pointer over the word in string
>
> 4) Goto2

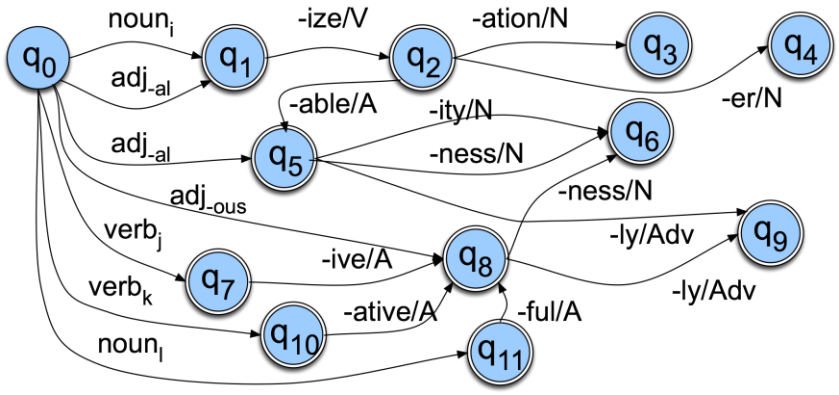
Strictly following the algorithm, you perhaps end up with failing to segment a string, if you cannot find a matching. For example, consider to segment the following string using the given lexicon.

String: thetablesdownthere

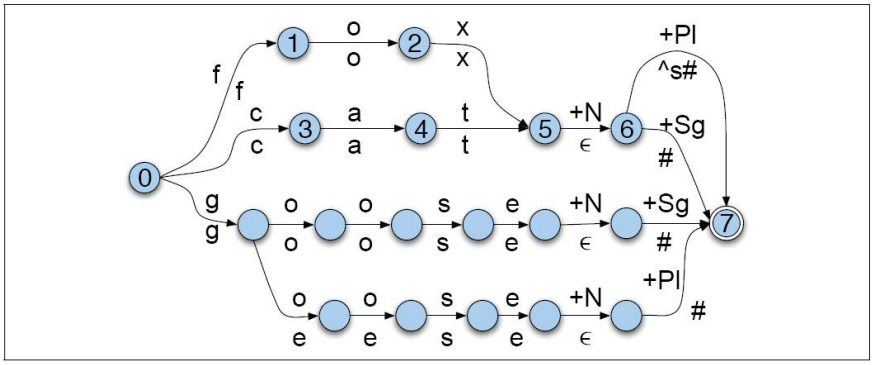
Lexicon: the table down there bled own.

Discuss how to fix the above problem.

Q2. Give examples of each of the noun and verb classes in the figure below, and find some exceptions to the rules.



Q3. Finite state transducer (FST) is a type of FSA that maps between *two sets of symbols*. The figure below is an FST that maps between surface level (*i.e.* actual spelling of words) and lexical level (*i.e.* concatenation of morphemes making up a word). For example, it can map "cats" and "cat+N+Pl" (Pl: plural).



Each transition is associated with a pair of two characters (*e.g.* e:o, #:+Pl). # indicates a word boundary, and ^ a morpheme boundary. Write a FST for the following mappings:

- cat – cat+N, cats – cat+N+Pl
- fox – fox+N, foxes – fox+N+Pl
- walk – walk+V, walked – walk+V+ed, walking – walk+V+ing
- stop – stop+V, stopped – stop+V+ed, stopping – stop+V+ing
- catch – catch+V, caught – catch+V+ed, catching – catch+V+ing

Q4. Write a program to do the following tasks:

1. Download the Web page of a given link and extract the text content of the page
2. Split the text into sentences and count sentences
3. Split the text into tokens and count token types
4. Find lemmas (or stems) of the tokens and count lemma types
5. Do stemming on the tokens and count unique 'stemmed' tokens

You may use *any* tools, including nltk, LingPipe, and Stanford NLP software.