# A Model Solution for Homework_3

YE Yuxiao

Tsinghua University

*yeyuxiao@outlook.com*

2017/10/27

# Clarification

- You need to **write some code** to finish the homework (unless specified otherwise)
- Please submit **Python** scripts, since we are learning Python in this course (other languages will not be accepted since next homework)
- File format: XXXX.**py** for scripts (other file formats will not be accepted since next homework)
- Try to add some documentation/comment

# Homework_3

Write a FST (a function) to implement the Soundex algorithm, which is explained as follows:

a. Keep the first letter of the name, and drop all occurrences of non-initial a, e, h, i, o, u, w, y

b. Replace the remaining letters with the following numbers:

   b, f, p, v $\rightarrow$ 1
   c, g, j, k, q, s, x, z $\rightarrow$ 2
   d, t $\rightarrow$ 3
   l $\rightarrow$ 4
   m, n $\rightarrow$ 5
   r $\rightarrow$ 6

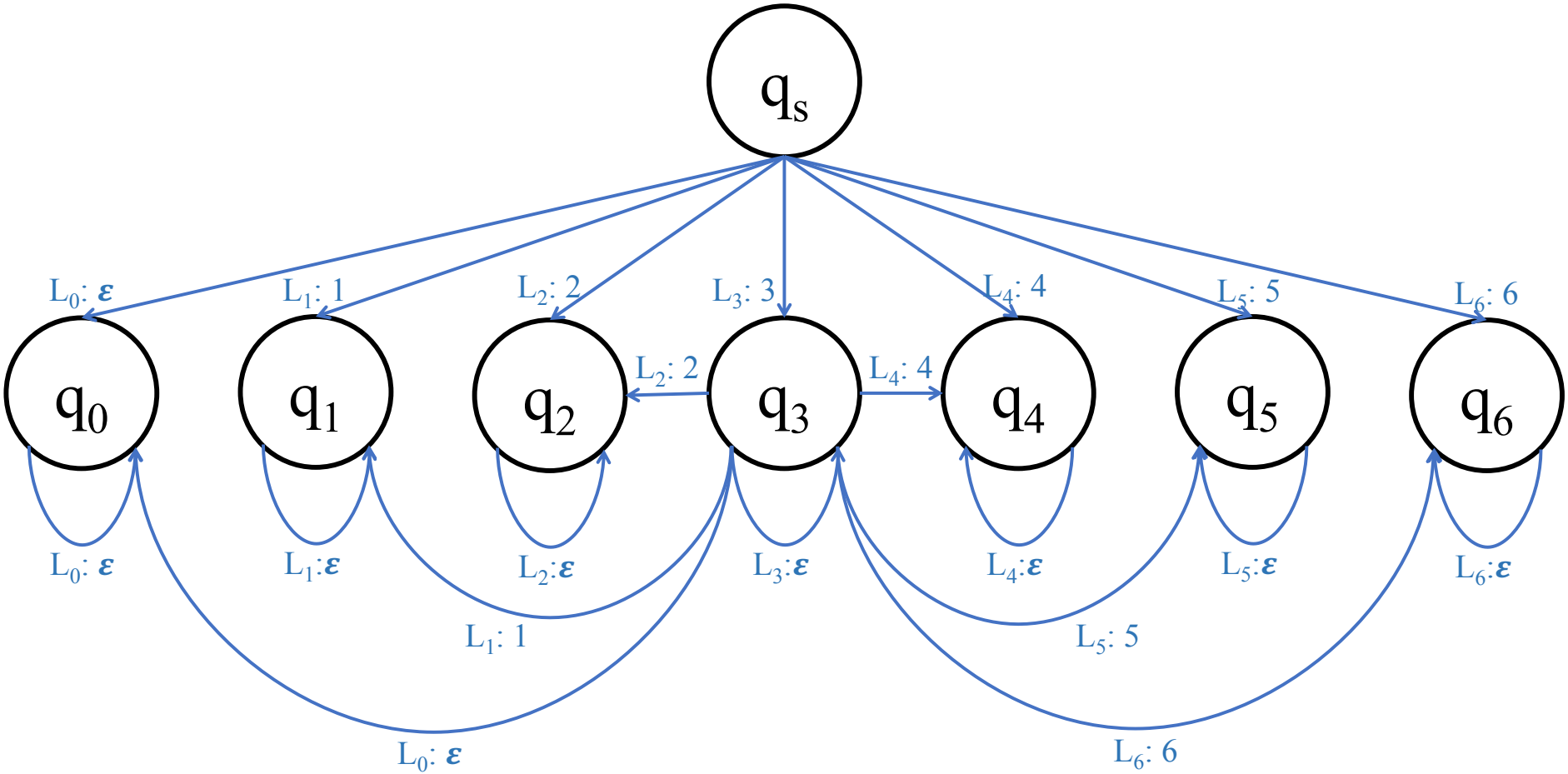
c. Replace any sequences of identical numbers , only if they derive from two or more letters that were adjacent in the original name, with a single number (e.g., 666 $\rightarrow$ 6)

d. Convert to the form Letter Digit Digit Digit (e.g., J612) by dropping digits past the third or padding with trailing zeros if necessary

# FST

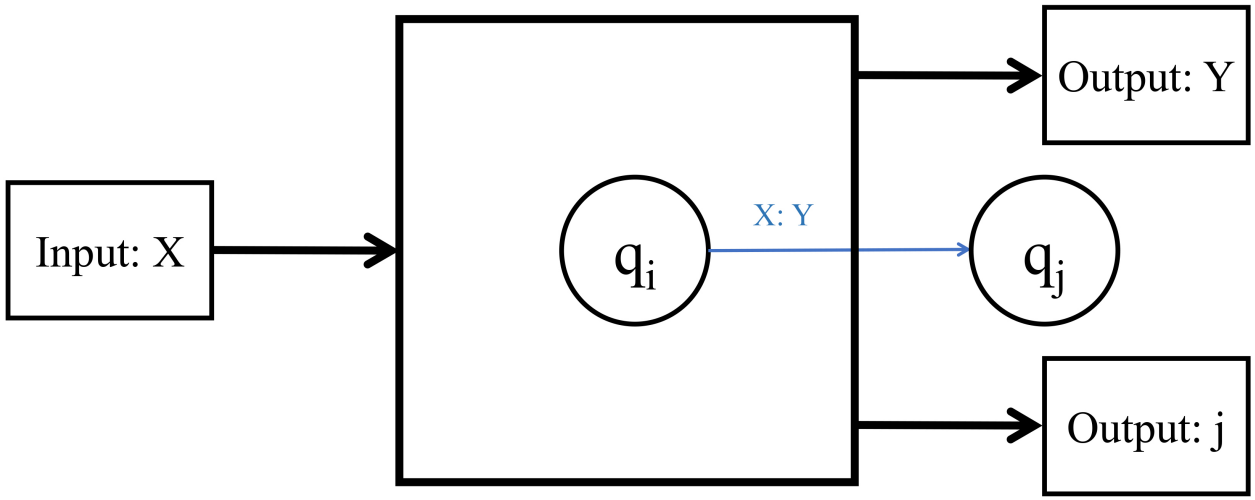
A simplified FST for the Soundex algorithm (without dropping or padding; final_state omitted)

- $L_0 = /[aehiouwy]/$, $L_1 = /[bfpv]/$, ... $L_6 = /r/$
- only shows full connections of $q_3$



# Abstraction of $q_i$

What $q_i$ is actually doing: takes as input a letter, outputs a letter (or $\epsilon$); and then goes to the next state.




- X: input letter
- Y: output digit
- i: index of current state
- j: index of next state