INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

Garden of Knowledge and Virtue

Date : 29-Nov-2020

[ASSIGNMENT - 1]

Natural Language Processing

COURSE CODE: CSC 4309, SEC: 01

LECTURER: Dr. SURIANI BT. SULAIMAN

Group Members :

MATRIC NO	NAME
1725413	HASAN TANVEER MAHMOOD
1721485	JAKI FAYEK ALVI RAHMAN
1725501	ISLAM MD RAISUL
1722279	LIU YUFEI

Code Snippet:

```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 16 22:53:17 2020

@author: Hasan Tanveer Mahmood 1725413
        Jaki Fayek Alvi Rahman 1721485
        Md Raisul Islam 1725501
        Liu Yufei 1722279
"""
import os
os.chdir('E:/')
from nltk.nltk_contrib.fst.fst import *

class myFST(FST):
   def recognize(self, iput, oput):

       self.inp = list(iput)
       self.outp = list(oput)

       if list(oput) == f.transduce(list(iput)):

          return True
       else:
          #print(outp)
          return False

f = myFST('german')

# declare the states
for i in range(1, 5):
   f.add_state(str(i))
f.initial_state = '1'
#f.initial_state = '2'
f.set_final('2')
f.set_final('4')

#setting up the arcs
f.add_arc('1', '2', ('0'), ('null'))
f.add_arc('1', '2', ('1'), ('eins'))
f.add_arc('1', '2', ('2'), ('zwei'))
f.add_arc('1', '2', ('3'), ('drei'))
f.add_arc('1', '2', ('4'), ('vier'))
f.add_arc('1', '2', ('5'), ('funf'))
f.add_arc('1', '2', ('6'), ('sechs'))
f.add_arc('1', '2', ('7'), ('sieben'))
f.add_arc('1', '2', ('8'), ('acht'))
```

```
f.add_arc('1', '2', ('9'), ('neun'))
f.add_arc('1', '2', ('10'), ('zehn'))
f.add_arc('1', '2', ('11'), ('elf'))
f.add_arc('1', '2', ('12'), ('zwolf'))
f.add_arc('1', '2', ('13'), ('drei-zehn'))
f.add_arc('1', '2', ('14'), ('vier-zehn'))
f.add_arc('1', '2', ('15'), ('funf-zehn'))
f.add_arc('1', '2', ('16'), ('sechs-zehn'))
f.add_arc('1', '2', ('17'), ('sieb-zehn'))
f.add_arc('1', '2', ('18'), ('acht-zehn'))
f.add_arc('1', '2', ('19'), ('neun-zehn'))
f.add_arc('1', '2', ('20'), ('zwan-zig'))
f.add_arc('1', '2', ('30'), ('drei-Big')) #general ides' don't support different letter formats like
Beta/Alpha
f.add_arc('1', '2', ('40'), ('vier-zig'))
f.add_arc('1', '2', ('50'), ('funf-zig'))
f.add_arc('1', '2', ('60'), ('sech-zig'))
f.add_arc('1', '2', ('70'), ('sieb-zig'))
f.add_arc('1', '2', ('80'), ('acht-zig'))
f.add_arc('1', '2', ('90'), ('neun-zig'))
f.add_arc('1', '2', ('100'), ('ein-hundert'))
f.add_arc('1', '2', ('1000'), ('ein-tausend'))

#Below section is for when there is input not Multiples of 10 (i.e. 20,30,40,....,100) and to add infix
and postfixes of the numeral input
f.add_arc('2', '3', (), ('-und-'))
f.add_arc('3', '4', ('20'), ('zwan-zig'))
f.add_arc('3', '4', ('30'), ('drei-Big'))
f.add_arc('3', '4', ('40'), ('vier-zig'))
f.add_arc('3', '4', ('50'), ('funf-zig'))
f.add_arc('3', '4', ('60'), ('sech-zig'))
f.add_arc('3', '4', ('70'), ('sieb-zig'))
f.add_arc('3', '4', ('80'), ('acht-zig'))
f.add_arc('3', '4', ('90'), ('neun-zig'))


#function to check the arcs for availability and writing to file
def translator(inp,outp):
   arcs_file = open('German-Trans.dat', 'a')
   arcs = ""
   arcs += ''.join(inp) + "  -->  "
   if(int(inp) >= 21):
      if (int(inp)%10 != 0 ):
         inp=int(inp)
         pref = inp%10
         postf = inp//10 * 10
         inp = str(pref) + str(postf)
```

```
    if f.recognize(inp, outp):
        print(outp)
        print("accept")
        arcs += ''.join(outp) + '\n'
    else:
        print("reject")
        arcs += ''.join('reject') + '\n'
    arcs_file.write(arcs)

inp = input('Enter the decimal Input: ')
outp = input('Enter the expected "German" Output: ')
print(inp)

#calling the function
translator(inp,outp)

#displaying the fst structure
disp = FSTDisplay(f)
```

Output Snap:

```
Enter the decimal Input: 10

Enter the expected "German" Output: zehn
10
zehn
accept

In [33]:
```

German-Trans.dat - Notepad

File Edit Format View Help

```
1  -->  eins
10  -->  zehn
13  -->  drei-zehn
20  -->  zwan-zig
30  -->  drei-Big
41  -->  eins-und-vier-zig
65  -->  funf-und-sech-zig
77  -->  sieben-und-sieb-zig
93  -->  drei-und-neun-zig
1000  -->  ein-tausend
101  -->  reject
```

Figure 1 : Number mapping to German numerals

FST construction snap:

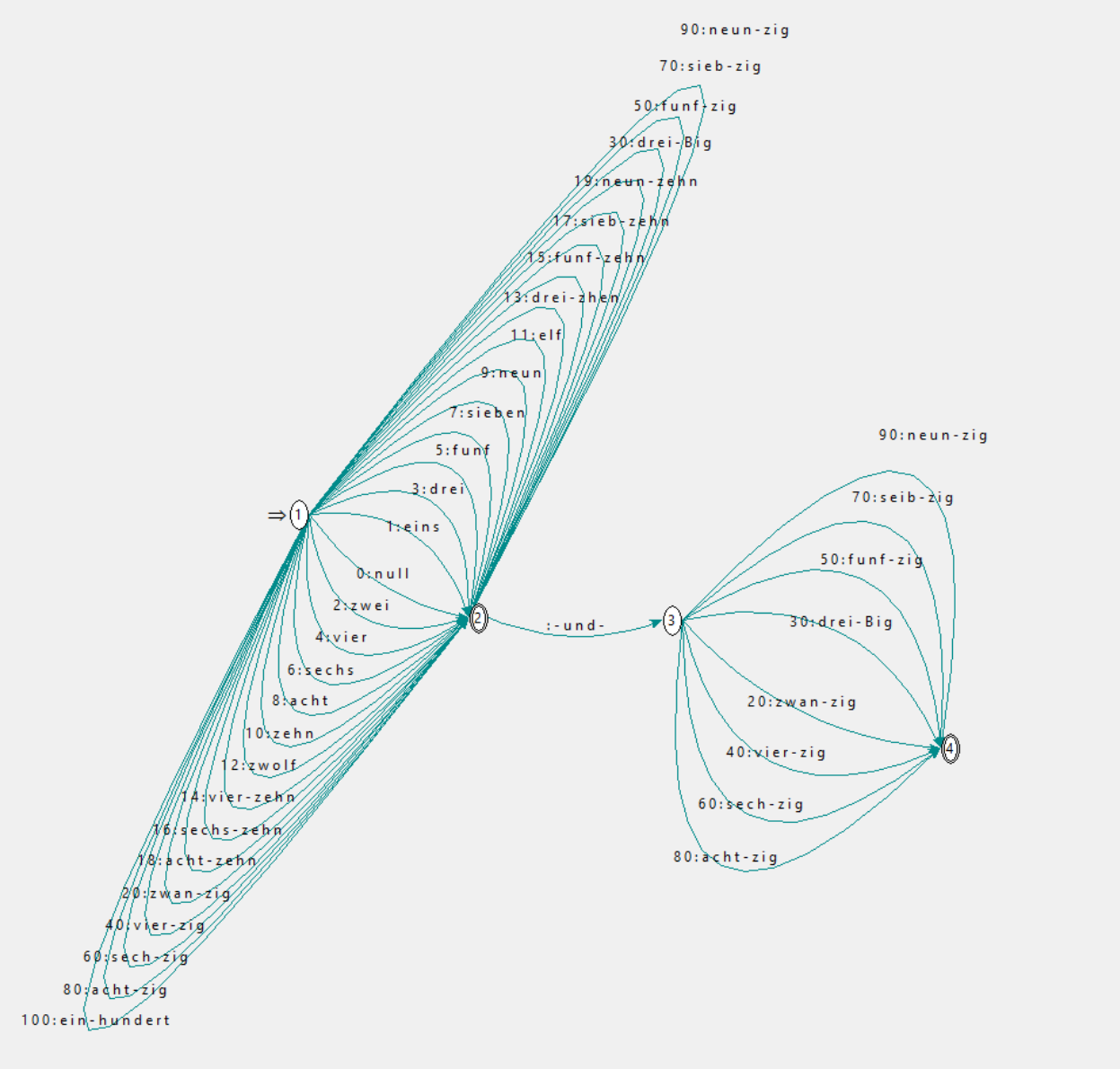

Figure 2 : FST Structure