ANLP 662 – Homework 5

Ekraam Sabir
USC-ID: 1428-2892-90
esabir@usc.edu

B. Tree/String Data

Q1. 4015 english tokens

Q2. 4363 spanish tokens

Q3. 4477 alignment links are present

Q4. 0.4483% of English tokens are unaligned

C. Writing a tree transducer

Q6.

```
head -1 hw5.train.trees | tiburon -k 1 - eng-spa.tt.manual
This is Tiburon, version 1.0
"Enumerar" "los" "vuelos" "de" "Baltimore" "a" "Seattle" "que" "se" "detienen" "en" "Minneapolis" "." # 1.000000

head -1 hw5.train.strings.spa | tiburon -k 1 eng-spa.tt.manual -
This is Tiburon, version 1.0
NP(VBP("List") NP(VB("the") TO("flights")) NP(VBP("from") VB("Baltimore") DT("to") TO("Seattle"))
NP(VB("that") WDT("stop") IN("in") VP(TO("Minneapolis") NNS("."))))) # 1.000000
```

D. Learning a tree transducer

Q9. 1504 distinct rules were extracted from the entire

Q10. 5 most commonly seen rules with counts are

466 : q.PP(x0:IN x1:NP) -> q.x0 q.x1
452: q.NP(x0:NNP) -> q.x0
346: q.PUNC(x0:".") -> t.x0
243: q.TO(x0:"to") -> t.x0
219: q.IN(x0:"from") -> t.x0

Q11.

The NP rules apparently can have so many variety of children:

q.NP(x0:PDT x1:DT x2:NNS) -> q.x0 q.x1 q.x2 # 0.00192061459667
q.NP(x0:CD x1:NNS) -> q.x0 q.x1 # 0.00192061459667
q.NP(x0:DT x1:ADJP x2:NN) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:PP x2:VP x3:VP) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:WDT x1:NNS) -> q.x0 q.x1 # 0.000640204865557
q.NP(x0:JJ x1:NN x2:NNS) -> q.x0 q.x1 q.x2 # 0.00384122919334
q.NP(x0:DT x1:NN x2:NN x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111
q.NP(x0:DT x1:NNP x2:NNP x3:NNP x4:NNS) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NNP x1:NNS) -> q.x0 q.x1 # 0.000640204865557
q.NP(x0:NNP x1:NN) -> q.x0 q.x1 # 0.0089628681178
q.NP(x0:NNP x1:NNP) -> q.x0 q.x1 # 0.0851472471191
q.NP(x0:NN x1:NN x2:NNS) -> q.x0 q.x1 q.x2 # 0.00320102432778
q.NP(x0:DT x1:NN) -> q.x0 q.x1 # 0.044814340589
q.NP(x0:NNP x1:CD x2:CD) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:CC x2:ADVP x3:NP) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:NN x1:NN x2:SYM) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:PRP) -> q.x0 # 0.0870678617157
q.NP(x0:CD x1:NNS x2:PP) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:PP) -> q.x0 q.x1 # 0.0345710627401
q.NP(x0:NNS) -> q.x0 # 0.0441741357234
q.NP(x0:NNP) -> q.x0 # 0.289372599232
q.NP(x0:DT x1:NNP x2:NNP x3:NNS) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111
q.NP(x0:NP x1:NN x2:CD x3:CD x4:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NP x1:PP x2:PP x3:ADVP) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:JJS x1:JJ x2:NN x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.00256081946223
q.NP(x0:DT x1:NN x2:SYM x3:CD) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:QP x1:RB) -> q.x0 q.x1 # 0.00192061459667
q.NP(x0:NP x1:SBAR x2:PP) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NNP x1:NNP x2:NNP) -> q.x0 q.x1 q.x2 # 0.0166453265045
q.NP(x0:DT x1:JJS x2:NN x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111
q.NP(x0:DT x1:JJS x2:NN) -> q.x0 q.x1 q.x2 # 0.0044814340589
q.NP(x0:NP x1:SBAR) -> q.x0 q.x1 # 0.0089628681178
q.NP(x0:CD x1:RB x2:RB) -> q.x0 q.x1 q.x2 # 0.00256081946223
q.NP(x0:NNP x1:NNP x2:NN x3:CD x4:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.00128040973111
q.NP(x0:DT x1:JJS x2:JJ x3:NN x4:NNS) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:DT x1:CD x2:CD x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:CD x1:NN x2:NNS) -> q.x0 q.x1 q.x2 # 0.00128040973111
q.NP(x0:RB x1:DT x2:NNP x3:NNP x4:NNS) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NP x1:VP) -> q.x0 q.x1 # 0.0044814340589
q.NP(x0:JJS x1:NN) -> q.x0 q.x1 # 0.00768245838668
q.NP(x0:NP x1:PP x2:PP x3:PP x4:PP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.00128040973111
q.NP(x0:NP x1:PP x2:PP x3:SBAR) -> q.x0 q.x1 q.x2 q.x3 # 0.00320102432778
q.NP(x0:NP x1:PP x2:PP x3:ADVP x4:NP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:JJS x1:NNS) -> q.x0 q.x1 # 0.000640204865557
q.NP(x0:DT x1:JJ x2:NNS) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NNP x1:CD x2:CD x3:CD) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:JJS x1:NN x2:NN) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NN x1:NNP x2:NNP x3:CD x4:CD x5:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 q.x5 # 0.000640204865557
q.NP(x0:QP x1:NNS) -> q.x0 q.x1 # 0.0044814340589
q.NP(x0:QP x1:NNS x2:QP) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:EX) -> q.x0 # 0.0128040973111
q.NP(x0:NN x1:NN x2:NNP x3:NNP x4:CD x5:CD x6:CD x7:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 q.x5 q.x6 q.x7 #

0.000640204865557
q.NP(x0:NP x1:ADJP) -> q.x0 q.x1 # 0.00128040973111
q.NP(x0:DT x1:NN x2:NN x3:SYM) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:CD x1:NN) -> q.x0 q.x1 # 0.00640204865557
q.NP(x0:DT x1:JJS x2:NNS) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:DT x1:JJ x2:NN) -> q.x0 q.x1 q.x2 # 0.00576184379001
q.NP(x0:NP x1:NP) -> q.x0 q.x1 # 0.0102432778489
q.NP(x0:NP x1:NP x2:PP x3:PP x4:PP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:DT x1:CD x2:CD x3:CD x4:NN) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NP x1:PP x2:PP) -> q.x0 q.x1 q.x2 # 0.0550576184379
q.NP(x0:CD x1:CD) -> q.x0 q.x1 # 0.00128040973111
q.NP(x0:NP x1:VP x2:SBAR) -> q.x0 q.x1 q.x2 # 0.00128040973111
q.NP(x0:NN x1:NNP x2:NNP x3:CD x4:CD x5:CD x6:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 q.x5 q.x6 #
0.00128040973111
q.NP(x0:NP x1:NP x2:PP x3:X x4:PP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NP x1:PP x2:PP x3:PP x4:NP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:CD x1:RB) -> q.x0 q.x1 # 0.0134443021767
q.NP(x0:DT x1:JJ) -> q.x0 q.x1 # 0.00320102432778
q.NP(x0:NP x1:VP x2:ADVP) -> q.x0 q.x1 q.x2 # 0.00192061459667
q.NP(x0:DT x1:NN x2:NN) -> q.x0 q.x1 q.x2 # 0.00512163892446
q.NP(x0:DT x1:NN x2:NNS) -> q.x0 q.x1 q.x2 # 0.00384122919334
q.NP(x0:NN) -> q.x0 # 0.0339308578745
q.NP(x0:DT x1:JJ x2:NN x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.00256081946223
q.NP(x0:NNP x1:JJ) -> q.x0 q.x1 # 0.00128040973111
q.NP(x0:NN x1:NNP x2:NNP) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:CC x2:PP x3:PP) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:DT x1:NN x2:SYM x3:SYM x4:SYM x5:CD x6:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 q.x5 q.x6 #
0.000640204865557
q.NP(x0:NN x1:NN) -> q.x0 q.x1 # 0.00768245838668
q.NP(x0:NP) -> q.x0 # 0.0128040973111
q.NP(x0:JJ x1:JJ x2:NN) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:PP x2:PP x3:NP x4:ADVP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NNP x1:POS) -> q.x0 q.x1 # 0.00128040973111
q.NP(x0:DT x1:NNP x2:NNP x3:CD x4:CD x5:NN) -> q.x0 q.x1 q.x2 q.x3 q.x4 q.x5 # 0.000640204865557
q.NP(x0:CD) -> q.x0 # 0.000640204865557
q.NP(x0:NP x1:PP x2:PP x3:PP) -> q.x0 q.x1 q.x2 q.x3 # 0.0102432778489
q.NP(x0:NP x1:PP x2:PP x3:PP x4:VP) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:CD x1:CD x2:CD x3:RB) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:NP x1:NNS) -> q.x0 q.x1 # 0.000640204865557
q.NP(x0:QP x1:NN) -> q.x0 q.x1 # 0.00128040973111
q.NP(x0:DT x1:NX) -> q.x0 q.x1 # 0.000640204865557
q.NP(x0:JJ x1:NNP x2:NNS) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:PP x2:PP x3:NP) -> q.x0 q.x1 q.x2 q.x3 # 0.00512163892446
q.NP(x0:CD x1:CD x2:NNS) -> q.x0 q.x1 q.x2 # 0.00192061459667
q.NP(x0:NP x1:NP x2:PP x3:PP) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111
q.NP(x0:DT) -> q.x0 # 0.00960307298335
q.NP(x0:NP x1:PP x2:PP x3:VP) -> q.x0 q.x1 q.x2 q.x3 # 0.00576184379001
q.NP(x0:DT x1:CD x2:NN x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:NN x1:NNP x2:NNP x3:CD x4:CD x5:CD x6:CD x7:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 q.x5 q.x6 q.x7 #
0.000640204865557
q.NP(x0:CC x1:NNP x2:CC x3:NNP) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:NP x1:PP x2:ADJP) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:DT x1:NNP x2:NNP x3:NN) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:JJS x1:NN x2:NNS) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NN x1:NNS) -> q.x0 q.x1 # 0.00512163892446

q.NP(x0:CD x1:CC x2:CD x3:RB) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111
q.NP(x0:DT x1:NNP x2:NN) -> q.x0 q.x1 q.x2 # 0.00128040973111
q.NP(x0:JJ x1:NN x2:NN) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:DT x1:NNS) -> q.x0 q.x1 # 0.0384122919334
q.NP(x0:NN x1:NNP x2:NNP x3:CD x4:CD) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:JJ x1:NNS) -> q.x0 q.x1 # 0.00384122919334
q.NP(x0:JJ x1:NNP) -> q.x0 q.x1 # 0.00192061459667
q.NP(x0:NP x1:PP x2:NP) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:NP x1:CC x2:NP) -> q.x0 q.x1 q.x2 # 0.0128040973111
q.NP(x0:NN x1:NN x2:NN) -> q.x0 q.x1 q.x2 # 0.000640204865557
q.NP(x0:JJ x1:NN) -> q.x0 q.x1 # 0.00704225352113
q.NP(x0:NNPS) -> q.x0 # 0.000640204865557
q.NP(x0:NN x1:CD x2:CD x3:CD) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111
q.NP(x0:DT x1:JJ x2:X x3:NNS) -> q.x0 q.x1 q.x2 q.x3 # 0.000640204865557
q.NP(x0:JJ x1:NN x2:JJ x3:NN x4:NN) -> q.x0 q.x1 q.x2 q.x3 q.x4 # 0.000640204865557
q.NP(x0:NNP x1:CC x2:NNP) -> q.x0 q.x1 q.x2 # 0.00128040973111
q.NP(x0:CD x1:CD x2:CD) -> q.x0 q.x1 q.x2 # 0.00192061459667
q.NP(x0:DT x1:JJ x2:NN x3:NNS) -> q.x0 q.x1 q.x2 q.x3 # 0.00128040973111

Q12. 1 derivation

$ head -1 hw5.train.trees | tiburon -k 1 - eng-spa.tt
This is Tiburon, version 1.0
"Enumerar" "los" "vuelos" "de" "Baltimore" "a" "Seattle" "que" "se" "detienen" "en" "Minneapolis" "." # 0.000003
$ head -1 hw5.train.trees | tiburon -c - eng-spa.tt
This is Tiburon, version 1.0
Tree info for input tree -:
    39 nodes
    12 yield length
Transducer info for input string transducer eng-spa.tt:
    2 states
    35 rules
CFG info for forward application of tree:
    39 states
    39 rules
    13 unique terminal symbols
    1 derivations

Q13. 3027024 derivations are possible

Top 5:

head -1 hw5.train.trees | tiburon -k 5 - eng-spa.tt
This is Tiburon, version 1.0
"Enumerar" "el" "vuelos" "de" "Baltimore" "a" "Seattle" "que" "parada" "en" "Minneapolis" "." # 2.491414E-24
"Enumerar" "los" "vuelos" "de" "Baltimore" "a" "Seattle" "que" "parada" "en" "Minneapolis" "." # 2.360287E-24
"Enumerar" "el" "vuelos" "desde" "Baltimore" "a" "Seattle" "que" "parada" "en" "Minneapolis" "." # 1.993131E-24
"Enumerar" "los" "vuelos" "desde" "Baltimore" "a" "Seattle" "que" "parada" "en" "Minneapolis" "." # 1.88823E-24
"Enumerar" "la" "vuelos" "de" "Baltimore" "a" "Seattle" "que" "parada" "en" "Minneapolis" "." # 1.376834E-24

Q14. There are infinite derivations possible for this.

Top 5:

head -1 hw5.train.strings.spa | tiburon -k 5 eng-spa.tt -
This is Tiburon, version 1.0
TOP(S(VP(VB("List") NP(NP(DT("the") NNS("flights")) PP(IN("from") NP(NNP("Baltimore"))) PP(TO("to") NP(NNP("Seattle"))) SBAR(WHNP(WDT("that")) S(VP(VBP("stop") PP(IN("in") NP(NNP("Minneapolis")))))))) PUNC(".")) # 7.867624E-25
TOP(S(VP(VB("List") NP(NP(DT("the") NNS("flights")) PP(IN("from") NP(NNP("Baltimore"))) PP(TO("to") NP(NNP("Seattle")))) NP(NP(DT("that") NN("stop")) PP(IN("in") NP(NNP("Minneapolis"))))) PUNC(".")) # 4.481606E-25
TOP(S(VP(VB("List") NP(NP(DT("the") NNS("flights")) PP(IN("from") NP(NNP("Baltimore"))) PP(TO("to") NP(NNP("Seattle"))) SBAR(WHNP(WDT("that")) S(VP(VBP("stop") PP(IN("on") NP(NNP("Minneapolis")))))))) PUNC(".")) # 3.639359E-25
TOP(S(VP(VB("List") NP(NP(DT("the") NNS("flights")) PP(IN("from") NP(NNP("Baltimore"))) PP(TO("to") NP(NNP("Seattle")))) NP(NP(DT("that") NN("stop")) PP(IN("on") NP(NNP("Minneapolis"))))) PUNC(".")) # 2.073075E-25
TOP(S(VP(VB("List") NP(NP(DT("the") NNS("flights")) PP(IN("from") NP(NNP("Baltimore"))) PP(TO("to") NP(NNP("Seattle"))) SBAR(WHNP(WDT("that")) S(VP(VB("stop") PP(IN("in") NP(NNP("Minneapolis")))))))) PUNC(".")) # 1.216737E-25