

NLP 作业六 命名实体识别

一. 使用 CRF++的运行语句与结果截图

1. 运行语句

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.19041.685]
(c) 2020 Microsoft Corporation. 保留所有权利。

C:\Users\Kuroko>cd 6

C:\Users\Kuroko\6>crf_learn template train.data model >> model_out.txt

C:\Users\Kuroko\6>crf_test -m model test.data >> result.txt

C:\Users\Kuroko\6>_
```

训练结果存放在 model_out.txt 中

测试结果存放在 result.txt 中

2. 训练结果：model_out.txt

```
model_out.txt
model_out.txt
1  CRF++: Yet Another CRF Tool Kit
2  Copyright (C) 2005-2013 Taku Kudo, All rights reserved.
3
4  reading training data: 100.. 200.. 300.. 400.. 500.. 600.. 700.. 800.. 900.. 1000.. 1100.. 1200.. 1300.. 1400.. 1500.. 1600
5  Done!5.56 s
6
7  Number of sentences: 12600
8  Number of features: 3318770
9  Number of thread(s): 4
10 Freq: 1
11 eta: 0.00010
12 C: 1.00000
13 shrinking size: 20
14 iter=0 terr=0.96394 serr=0.99889 act=3318770 obj=413617.96542 diff=1.00000
15 iter=1 terr=0.16959 serr=0.79540 act=3318770 obj=264607.52245 diff=0.36026
16 iter=2 terr=0.16959 serr=0.79540 act=3318770 obj=222201.37286 diff=0.16026
17 iter=3 terr=0.16959 serr=0.79540 act=3318770 obj=144778.79585 diff=0.34843
18 iter=4 terr=0.61520 serr=0.97024 act=3318770 obj=1451034.80654 diff=9.02243
19 iter=5 terr=0.13740 serr=0.74365 act=3318770 obj=94872.40155 diff=0.93462
20 iter=6 terr=0.14522 serr=0.75810 act=3318770 obj=84248.16939 diff=0.11198
21 iter=7 terr=0.13376 serr=0.73643 act=3318770 obj=74762.35640 diff=0.11259
22 iter=8 terr=0.11613 serr=0.66294 act=3318770 obj=58325.06710 diff=0.21986
23 iter=9 terr=0.10197 serr=0.59151 act=3318770 obj=51307.71947 diff=0.12031
24 iter=10 terr=0.09686 serr=0.57762 act=3318770 obj=47014.08520 diff=0.08368
25 iter=11 terr=0.09104 serr=0.54056 act=3318770 obj=42493.79409 diff=0.09615
26 iter=12 terr=0.08683 serr=0.52651 act=3318770 obj=39667.97088 diff=0.06650
27 iter=13 terr=0.08414 serr=0.50571 act=3318770 obj=39454.74475 diff=0.00538
28 iter=14 terr=0.07913 serr=0.47960 act=3318770 obj=36689.55463 diff=0.07009
29 iter=15 terr=0.07824 serr=0.47119 act=3318770 obj=35964.74375 diff=0.01976
30 iter=16 terr=0.07790 serr=0.47317 act=3318770 obj=34950.62779 diff=0.02820
```

model_out.txt X

model_out.txt

```
31 iter=17 terr=0.07476 serr=0.45579 act=3318770 obj=33429.05276 diff=0.04353
32 iter=18 terr=0.07283 serr=0.44976 act=3318770 obj=331904.30037 diff=0.04561
33 iter=19 terr=0.06886 serr=0.43365 act=3318770 obj=30446.81316 diff=0.04568
34 iter=20 terr=0.06610 serr=0.42437 act=3318770 obj=29647.16559 diff=0.02626
35 iter=21 terr=0.06363 serr=0.41833 act=3318770 obj=28529.89741 diff=0.03769
36 iter=22 terr=0.06118 serr=0.41143 act=3318770 obj=26794.51692 diff=0.06083
37 iter=23 terr=0.05739 serr=0.38794 act=3318770 obj=25372.11547 diff=0.05309
38 iter=24 terr=0.05596 serr=0.38310 act=3318770 obj=24721.92073 diff=0.02563
39 iter=25 terr=0.05237 serr=0.36413 act=3318770 obj=23785.42672 diff=0.03788
40 iter=26 terr=0.05454 serr=0.35365 act=3318770 obj=24164.09647 diff=0.01592
41 iter=27 terr=0.05012 serr=0.35135 act=3318770 obj=23185.16342 diff=0.04051
42 iter=28 terr=0.04761 serr=0.33905 act=3318770 obj=22338.20512 diff=0.03653
43 iter=29 terr=0.04639 serr=0.33024 act=3318770 obj=21354.97579 diff=0.04402
44 iter=30 terr=0.04498 serr=0.32730 act=3318770 obj=20634.78346 diff=0.03372
45 iter=31 terr=0.04289 serr=0.31683 act=3318770 obj=19825.76096 diff=0.03921
46 iter=32 terr=0.04003 serr=0.30238 act=3318770 obj=19160.80888 diff=0.03354
47 iter=33 terr=0.03777 serr=0.28857 act=3318770 obj=18192.25255 diff=0.05055
48 iter=34 terr=0.03582 serr=0.27421 act=3318770 obj=17606.50793 diff=0.03220
49 iter=35 terr=0.03412 serr=0.26325 act=3318770 obj=16996.73802 diff=0.03463
50 iter=36 terr=0.03282 serr=0.26032 act=3318770 obj=16447.27153 diff=0.03233
51 iter=37 terr=0.03138 serr=0.25317 act=3318770 obj=16027.87163 diff=0.02550
52 iter=38 terr=0.03144 serr=0.25619 act=3318770 obj=15780.56321 diff=0.01543
53 iter=39 terr=0.02819 serr=0.23548 act=3318770 obj=15085.70030 diff=0.04403
54 iter=40 terr=0.02728 serr=0.22873 act=3318770 obj=14820.88856 diff=0.01755
55 iter=41 terr=0.02639 serr=0.22190 act=3318770 obj=14456.22789 diff=0.02460
56 iter=42 terr=0.02542 serr=0.21913 act=3318770 obj=14090.02932 diff=0.02533
57 iter=43 terr=0.02392 serr=0.20968 act=3318770 obj=13694.28441 diff=0.02809
58 iter=44 terr=0.02280 serr=0.20143 act=3318770 obj=13333.71410 diff=0.02633
59 iter=45 terr=0.02182 serr=0.19548 act=3318770 obj=13000.00525 diff=0.02503
60 iter=46 terr=0.02149 serr=0.19325 act=3318770 obj=12584.61229 diff=0.03195
61 iter=47 terr=0.01965 serr=0.17667 act=3318770 obj=12116.57215 diff=0.03719
62 iter=48 terr=0.01823 serr=0.16754 act=3318770 obj=11826.23789 diff=0.02396
63 iter=49 terr=0.01662 serr=0.15603 act=3318770 obj=11451.16451 diff=0.03172
64 iter=50 terr=0.01699 serr=0.16357 act=3318770 obj=11442.32820 diff=0.00077
65 iter=51 terr=0.01583 serr=0.15413 act=3318770 obj=11243.48387 diff=0.01738
66 iter=52 terr=0.01491 serr=0.14706 act=3318770 obj=11007.44796 diff=0.02099
67 iter=53 terr=0.01325 serr=0.13175 act=3318770 obj=10557.35877 diff=0.04089
68 iter=54 terr=0.01325 serr=0.13024 act=3318770 obj=10377.57389 diff=0.01703
69 iter=55 terr=0.01206 serr=0.12135 act=3318770 obj=10184.67091 diff=0.01859
70 iter=56 terr=0.01153 serr=0.11635 act=3318770 obj=10000.06893 diff=0.01813
```

model_out.txt X

model_out.txt

```
70 iter=56 terr=0.01153 serr=0.11635 act=3318770 obj=10000.06893 diff=0.01813
71 iter=57 terr=0.01064 serr=0.10786 act=3318770 obj=9815.57515 diff=0.01845
72 iter=58 terr=0.01017 serr=0.10627 act=3318770 obj=9660.53709 diff=0.01580
73 iter=59 terr=0.00958 serr=0.10198 act=3318770 obj=9541.13898 diff=0.01236
74 iter=60 terr=0.00893 serr=0.09563 act=3318770 obj=9349.12491 diff=0.02012
75 iter=61 terr=0.00829 serr=0.08944 act=3318770 obj=9237.50434 diff=0.01194
76 iter=62 terr=0.00785 serr=0.08722 act=3318770 obj=9109.03383 diff=0.01391
77 iter=63 terr=0.00706 serr=0.08016 act=3318770 obj=8966.22634 diff=0.01568
78 iter=64 terr=0.00646 serr=0.07381 act=3318770 obj=8817.85991 diff=0.01655
79 iter=65 terr=0.00591 serr=0.06905 act=3318770 obj=8729.75099 diff=0.00999
80 iter=66 terr=0.00571 serr=0.06619 act=3318770 obj=8660.72081 diff=0.00791
81 iter=67 terr=0.00562 serr=0.06468 act=3318770 obj=8594.80717 diff=0.00761
82 iter=68 terr=0.00515 serr=0.06032 act=3318770 obj=8533.96389 diff=0.00708
83 iter=69 terr=0.00521 serr=0.06079 act=3318770 obj=8472.48464 diff=0.00720
84 iter=70 terr=0.00490 serr=0.05817 act=3318770 obj=8420.98003 diff=0.00608
85 iter=71 terr=0.00447 serr=0.05405 act=3318770 obj=8324.33826 diff=0.01148
86 iter=72 terr=0.00389 serr=0.04810 act=3318770 obj=8250.78891 diff=0.00884
87 iter=73 terr=0.00362 serr=0.04460 act=3318770 obj=8179.08340 diff=0.00869
88 iter=74 terr=0.00333 serr=0.04127 act=3318770 obj=8113.80346 diff=0.00798
89 iter=75 terr=0.00294 serr=0.03675 act=3318770 obj=8062.27828 diff=0.00635
90 iter=76 terr=0.00275 serr=0.03484 act=3318770 obj=7986.30421 diff=0.00942
91 iter=77 terr=0.00256 serr=0.03270 act=3318770 obj=7949.87993 diff=0.00456
92 iter=78 terr=0.00249 serr=0.03198 act=3318770 obj=7909.73663 diff=0.00505
93 iter=79 terr=0.00210 serr=0.02667 act=3318770 obj=7874.14767 diff=0.00450
94 iter=80 terr=0.00210 serr=0.02698 act=3318770 obj=7830.54154 diff=0.00554
95 iter=81 terr=0.00199 serr=0.02571 act=3318770 obj=7789.24310 diff=0.00527
96 iter=82 terr=0.00188 serr=0.02452 act=3318770 obj=7759.66857 diff=0.00380
97 iter=83 terr=0.00150 serr=0.01976 act=3318770 obj=7720.43470 diff=0.00506
98 iter=84 terr=0.00122 serr=0.01643 act=3318770 obj=7674.89854 diff=0.00590
99 iter=85 terr=0.00112 serr=0.01508 act=3318770 obj=7651.78584 diff=0.00301
100 iter=86 terr=0.00106 serr=0.01429 act=3318770 obj=7621.52733 diff=0.00395
101 iter=87 terr=0.00102 serr=0.01373 act=3318770 obj=7591.41235 diff=0.00395
102 iter=88 terr=0.00093 serr=0.01254 act=3318770 obj=7552.84930 diff=0.00508
103 iter=89 terr=0.00081 serr=0.01095 act=3318770 obj=7501.55437 diff=0.00679
104 iter=90 terr=0.00080 serr=0.01056 act=3318770 obj=7466.64011 diff=0.00465
105 iter=91 terr=0.00066 serr=0.00897 act=3318770 obj=7423.81527 diff=0.00574
106 iter=92 terr=0.00065 serr=0.00873 act=3318770 obj=7395.21463 diff=0.00385
107 iter=93 terr=0.00056 serr=0.00762 act=3318770 obj=7363.40959 diff=0.00430
108 iter=94 terr=0.00062 serr=0.00825 act=3318770 obj=7338.08886 diff=0.00344
109 iter=95 terr=0.00056 serr=0.00746 act=3318770 obj=7307.20995 diff=0.00421
110 iter=96 terr=0.00058 serr=0.00778 act=3318770 obj=7279.49509 diff=0.00379
```

```
model_out.txt X
model_out.txt
120 iter=106 terr=0.00024 serr=0.00317 act=3318770 obj=7084.74214 diff=0.00174
121 iter=107 terr=0.00026 serr=0.00349 act=3318770 obj=7070.40249 diff=0.00202
122 iter=108 terr=0.00025 serr=0.00333 act=3318770 obj=7059.81819 diff=0.00150
123 iter=109 terr=0.00022 serr=0.00286 act=3318770 obj=7036.46692 diff=0.00331
124 iter=110 terr=0.00019 serr=0.00246 act=3318770 obj=7021.82410 diff=0.00208
125 iter=111 terr=0.00016 serr=0.00214 act=3318770 obj=7002.31301 diff=0.00278
126 iter=112 terr=0.00014 serr=0.00183 act=3318770 obj=6991.53504 diff=0.00154
127 iter=113 terr=0.00015 serr=0.00198 act=3318770 obj=6982.47504 diff=0.00130
128 iter=114 terr=0.00014 serr=0.00190 act=3318770 obj=6975.57617 diff=0.00099
129 iter=115 terr=0.00015 serr=0.00198 act=3318770 obj=6967.24389 diff=0.00119
130 iter=116 terr=0.00016 serr=0.00206 act=3318770 obj=6958.51984 diff=0.00125
131 iter=117 terr=0.00016 serr=0.00206 act=3318770 obj=6953.06328 diff=0.00078
132 iter=118 terr=0.00017 serr=0.00230 act=3318770 obj=6941.60570 diff=0.00165
133 iter=119 terr=0.00019 serr=0.00254 act=3318770 obj=6931.06602 diff=0.00152
134 iter=120 terr=0.00019 serr=0.00262 act=3318770 obj=6924.85488 diff=0.00090
135 iter=121 terr=0.00019 serr=0.00254 act=3318770 obj=6917.36992 diff=0.00108
136 iter=122 terr=0.00018 serr=0.00238 act=3318770 obj=6913.61046 diff=0.00054
137 iter=123 terr=0.00019 serr=0.00254 act=3318770 obj=6907.19479 diff=0.00093
138 iter=124 terr=0.00019 serr=0.00254 act=3318770 obj=6898.11057 diff=0.00132
139 iter=125 terr=0.00021 serr=0.00278 act=3318770 obj=6894.18296 diff=0.00057
140 iter=126 terr=0.00023 serr=0.00310 act=3318770 obj=6888.36737 diff=0.00084
141 iter=127 terr=0.00024 serr=0.00325 act=3318770 obj=6882.75613 diff=0.00081
142 iter=128 terr=0.00023 serr=0.00310 act=3318770 obj=6880.10124 diff=0.00039
143 iter=129 terr=0.00023 serr=0.00302 act=3318770 obj=6875.93599 diff=0.00061
144 iter=130 terr=0.00029 serr=0.00389 act=3318770 obj=6873.24737 diff=0.00039
145 iter=131 terr=0.00026 serr=0.00341 act=3318770 obj=6868.64589 diff=0.00067
146 iter=132 terr=0.00025 serr=0.00333 act=3318770 obj=6865.07970 diff=0.00052
147 iter=133 terr=0.00025 serr=0.00333 act=3318770 obj=6862.70826 diff=0.00035
148 iter=134 terr=0.00026 serr=0.00333 act=3318770 obj=6858.20755 diff=0.00066
149 iter=135 terr=0.00024 serr=0.00310 act=3318770 obj=6854.64760 diff=0.00052
150 iter=136 terr=0.00026 serr=0.00333 act=3318770 obj=6851.83021 diff=0.00041
151 iter=137 terr=0.00027 serr=0.00357 act=3318770 obj=6849.49378 diff=0.00034
152 iter=138 terr=0.00028 serr=0.00373 act=3318770 obj=6846.76304 diff=0.00040
153 iter=139 terr=0.00028 serr=0.00373 act=3318770 obj=6843.33189 diff=0.00050
154 iter=140 terr=0.00028 serr=0.00365 act=3318770 obj=6840.89461 diff=0.00036
155 iter=141 terr=0.00028 serr=0.00373 act=3318770 obj=6839.32255 diff=0.00023
156 iter=142 terr=0.00029 serr=0.00381 act=3318770 obj=6837.36544 diff=0.00029
157 iter=143 terr=0.00028 serr=0.00365 act=3318770 obj=6835.92877 diff=0.00021
158 iter=144 terr=0.00028 serr=0.00373 act=3318770 obj=6834.21614 diff=0.00025
159 iter=145 terr=0.00030 serr=0.00397 act=3318770 obj=6833.45204 diff=0.00011
model_out.txt X
model_out.txt
157 iter=143 terr=0.00028 serr=0.00365 act=3318770 obj=6835.92877 diff=0.00021
158 iter=144 terr=0.00028 serr=0.00373 act=3318770 obj=6834.21614 diff=0.00025
159 iter=145 terr=0.00030 serr=0.00397 act=3318770 obj=6833.45204 diff=0.00011
160 iter=146 terr=0.00032 serr=0.00421 act=3318770 obj=6832.42456 diff=0.00015
161 iter=147 terr=0.00029 serr=0.00373 act=3318770 obj=6831.29763 diff=0.00016
162 iter=148 terr=0.00029 serr=0.00373 act=3318770 obj=6829.84322 diff=0.00021
163 iter=149 terr=0.00028 serr=0.00365 act=3318770 obj=6828.79646 diff=0.00015
164 iter=150 terr=0.00029 serr=0.00373 act=3318770 obj=6827.79845 diff=0.00015
165 iter=151 terr=0.00028 serr=0.00373 act=3318770 obj=6827.04276 diff=0.00011
166 iter=152 terr=0.00031 serr=0.00397 act=3318770 obj=6825.53970 diff=0.00022
167 iter=153 terr=0.00033 serr=0.00429 act=3318770 obj=6824.68375 diff=0.00013
168 iter=154 terr=0.00033 serr=0.00429 act=3318770 obj=6823.90246 diff=0.00011
169 iter=155 terr=0.00034 serr=0.00444 act=3318770 obj=6823.11089 diff=0.00012
170 iter=156 terr=0.00033 serr=0.00444 act=3318770 obj=6822.01388 diff=0.00016
171 iter=157 terr=0.00037 serr=0.00500 act=3318770 obj=6821.01202 diff=0.00015
172 iter=158 terr=0.00037 serr=0.00500 act=3318770 obj=6820.28564 diff=0.00011
173 iter=159 terr=0.00037 serr=0.00492 act=3318770 obj=6820.35593 diff=0.00001
174 iter=160 terr=0.00037 serr=0.00492 act=3318770 obj=6819.74495 diff=0.00009
175 iter=161 terr=0.00036 serr=0.00484 act=3318770 obj=6819.06681 diff=0.00010
176
177 Done!190.10 s
178
179
```

3. 测试结果：result.txt

≡ result.txt ×

≡ result.txt

```
1 3. NNP O O
2 Jeff NNP B-PER B-PER
3 Bean NNP I-PER I-PER
4 ( ( O O
5 Canada NNP B-LOC B-LOC
6 ) ) O O
7 209.96 CD O O
8
9 4. CD O O
10 Eric NNP B-PER B-PER
11 Bergoust NNP I-PER I-PER
12 ( ( O O
13 U.S NNP B-LOC B-LOC
14 ) ) O O
15 207.15 CD O O
16
17 5. CD O O
18 Christian NNP B-PER B-PER
19 Rijavec NNP I-PER I-PER
20 ( ( O O
21 Austria NNP B-LOC B-LOC
22 ) ) O O
23 204.17 CD O O
24
25 6. CD O O
26 Alexandre NNP B-PER B-PER
27 Mikhailov NNP I-PER I-PER
28 ( ( O O
29 Russia NNP B-LOC B-LOC
30 ) ) O O
31 202.59 CD O O
32
33 7. CD O O
34 Ales NNPS B-PER B-PER
35 Valenta NNP I-PER I-PER
36 ( ( O O
37 Czech NNP B-LOC B-LOC
38 Republic NNP I-LOC I-LOC
39 ) ) O O
40 194.02 CD O O
```

```
result.txt x
result.txt
22850
22851 NEW NNP B-ORG B-ORG
22852 YORK NNP I-ORG I-ORG
22853 59 CD 0 0
22854 69 CD 0 0
22855 .461 CD 0 0
22856 21 CD 0 0
22857 1/2 CD 0 0
22858
22859 FLORIDA NNP B-ORG B-ORG
22860 58 CD 0 0
22861 69 CD 0 0
22862 .457 CD 0 0
22863 22 CD 0 0
22864
22865 PHILADELPHIA NNP B-ORG B-ORG
22866 52 CD 0 0
22867 75 CD 0 0
22868 .409 CD 0 0
22869 28 CD 0 0
22870
22871 CENTRAL NNP B-MISC B-MISC
22872 DIVISION NNP I-MISC I-MISC
22873
22874 HOUSTON NNP B-ORG B-ORG
22875 68 CD 0 0
22876 59 CD 0 0
22877 .535 CD 0 0
22878 - : 0 0
22879
22880 ST PRP B-ORG B-ORG
22881 LOUIS VBZ I-ORG I-ORG
22882 67 CD 0 0
22883 59 CD 0 0
22884 .532 CD 0 0
22885 1/2 CD 0 0
22886
22887
```

二. 对 template 模板的注释理解

1. 模板宏

在模板中，每一行代表一个模板，而在每个模板中，通过`%x[row,col]`的宏来标记输入的数据。

<code>%x[0,0]</code>	the
<code>%x[0,1]</code>	DT
<code>%x[-1,0]</code>	reckons
<code>%x[-2,1]</code>	PRP
<code>%x[0,0]/%x[0,1]</code>	the/DT
<code>ABC%x[0,1]123</code>	ABCDT123

2. 模板分类

模板通过第一个字符的值表示类型。

2.1 Unigram 模板，通过第一个字符 ‘U’ 来标识

2.2 Bigram 模板，通过第一个字符 ‘B’ 来标识

3. 实例分析

本次 template 文件中给出的模板是 Unigram 类型，我们对此进行分析

```
U00:%x[-2, 0]
U01:%x[-1, 0]
U02:%x[0, 0]
U03:%x[1, 0]
U04:%x[2, 0]
U05:%x[-1, 0]/%x[0, 0]
U06:%x[0, 0]/%x[1, 0]

U10:%x[-2, 1]
U11:%x[-1, 1]
U12:%x[0, 1]
U13:%x[1, 1]
U14:%x[2, 1]
U15:%x[-2, 1]/%x[-1, 1]
U16:%x[-1, 1]/%x[0, 1]
U17:%x[0, 1]/%x[1, 1]
U18:%x[1, 1]/%x[2, 1]

U20:%x[-2, 1]/%x[-1, 1]/%x[0, 1]
U21:%x[-1, 1]/%x[0, 1]/%x[1, 1]
U22:%x[0, 1]/%x[1, 1]/%x[2, 1]
```

在本次测试文件 train.data 和 test.data 中，共有三列，第一列为单词本身，第二列为词性，而第三列为实体识别的标签。

注意到第三列为 B-XXX 与 I-XXX 的形式，B 表示表示一个词的开头那个字，I 表示一个词的结尾那个字，0 则会在 python 源代码中进行剔除。而 B 与 I 也是我们在 python 中划分单词的重要依据。

然后再关注到模板的前五行：

```
U00:%x[-2, 0]
U01:%x[-1, 0]
U02:%x[0, 0]
U03:%x[1, 0]
U04:%x[2, 0]
```

-2 到 2 表示了在处理每一行的时候，将关联其前两行与后两行所以以 train.data 的前五行（非 0 行）为例子

```
American NNP B-MISC
League NNP I-MISC
Cleveland NNP B-ORG
DETROIT NNP B-ORG
```

BALTIMORE VB B-ORG

第一行将生产如下函数:

```
func1=if(output=B-ORG and feature= "U00:American" )  
    return 1 else return 0
```

其中 output=B 指的是当前词的预测标记, 也就是 Cleveland 的预测标记,

```
func2=if(output=B-MISC and feature= "U00:American" )  
    return 1 else return 0
```

```
func3=if(output=B-PER and feature= "U00:American" )  
    return 1 else return 0
```

```
func4=if(output=B-LOC and feature= "U00:American" )  
    return 1 else return 0
```

.....

```
funcXX = if (output = B-NP and feature="U00:NN")  
    return 1  else return 0
```

```
funcXY = if (output = 0 and feature="U00:NN")  
    return 1  else return 0
```

模板生成的特征函数的数量总计为 $(L * N)$, 其中 L 是输出类的数量, N 是从给定模板扩展的唯一字符串的数量。

U00:%x[-2, 0]	-> American
U01:%x[-1, 0]	-> League
U02:%x[0, 0]	-> Cleveland
U03:%x[1, 0]	-> DETROIT
U04:%x[2, 0]	-> BALTIMORE
U05:%x[-1, 0]/%x[0, 0]	-> League/Cleveland
U06:%x[0, 0]/%x[1, 0]	-> Cleveland/DETROIT

U10:%x[-2, 1]	-> NNP
U11:%x[-1, 1]	-> NNP
U12:%x[0, 1]	-> NNP
U13:%x[1, 1]	-> NNP
U14:%x[2, 1]	-> VB
U15:%x[-2, 1]/%x[-1, 1]	-> NNP/NNP
U16:%x[-1, 1]/%x[0, 1]	-> NNP/NNP
U17:%x[0, 1]/%x[1, 1]	-> NNP/NNP
U18:%x[1, 1]/%x[2, 1]	-> NNP/VB

U20:%x[-2, 1]/%x[-1, 1]/%x[0, 1]	-> NNP/NNP/NNP
U21:%x[-1, 1]/%x[0, 1]/%x[1, 1]	-> NNP/NNP/NNP
U22:%x[0, 1]/%x[1, 1]/%x[2, 1]	-> NNP/NNP/VB

每个模板会把所有可能的标记输出都列一遍, 然后通过训练确定每种标记的

权重，合理的标记在训练样本中出现的次数多，对应的权重就高，不合理的标记在训练样本中出现的少，对应的权重就少，但是在利用模板生成转移特征函数是会把所有可能的特征函数都列出来，由模型通过训练决定每个特征的重要程度。