HW1-Observations
Name: Haoran Guo
UNI:hg2461

Question 4

```
~/PycharmProjects/Haoran/nlp_hw1 python eval_ne_tagger.py ner_dev.key 4_2.txt
Found 14043 NEs. Expected 5931 NEs; Correct: 3117.

         precision      recall          F1-Score
Total:   0.221961       0.525544        0.312106
PER:     0.435451       0.231230        0.302061
ORG:     0.475936       0.399103        0.434146
LOC:     0.147750       0.870229        0.252612
MISC:    0.491689       0.610206        0.544574
```

The naive tagger has a low recall rate and also poor at precision. F1 score shows that the current model is not a good choice for word tagging process. Recall rate for person and organization is very low. Moreover, the precision for location is poor that it is hard for us to trust the tagging output we've made.

Question 5

```
~/PycharmProjects/Haoran/nlp_hw1 python eval_ne_tagger.py ner_dev.key 5_2.txt
Found 4704 NEs. Expected 5931 NEs; Correct: 3647.

         precision      recall          F1-Score
Total:   0.775298       0.614905        0.685849
PER:     0.762535       0.595756        0.668907
ORG:     0.611855       0.478326        0.536913
LOC:     0.876458       0.696292        0.776056
MISC:    0.830065       0.689468        0.753262
```

The HMM tagger with a simple class _RARE_ for rare word performs much better than the one in question 4. The precision on tagging location words increased a lot, while the recall rate of person grows rapidly. Although the total precision reached 77.5%, the recall rate, which is about 61.5%, still needs to be improved, especially for person(PER) and organizations(ORG).

Question 6

```
~/PycharmProjects/Haoran/nlp_hw1 python eval_ne_tagger.py ner_dev.key 6.txt
Found 5844 NEs. Expected 5931 NEs; Correct: 4334.

         precision      recall          F1-Score
Total:   0.741615       0.730737        0.736136
PER:     0.808343       0.780196        0.794020
ORG:     0.533533       0.665919        0.592420
LOC:     0.842779       0.754089        0.795971
MISC:    0.824769       0.679696        0.745238
```

In this question, I splitted the original “_RARE_” class into several smaller classes.

As I went through the words in training set which were labeled as “_RARE_” in question 4, I found a lot of them were numeric, all-capitalized or words joint by “-”. Here are the classification criterias I made:

“_DASHED_NUM_”: Words with dash and only numeric characters, often date or phone numbers. For instance:”7-283”
“_DASHED_UPPER_”: Capitalized letters joint by dash, usually abbreviate names or organization names. For example: “NSDAP-AO”
“_DASHED_”: Other dashed words, often constructed by shorter words, like “five-nation”, “co-operation”, etc
“_CHAR_”: Words without alpha or numbers, often single “[”, “:” or “.”, etc
“_DIGIT_”: Words with only numeric characters, “.” and “,”
“_UPPER_”: Words with only capital letters
“_LOWER_”: Words with only lower case letters, often regular words. Like “golden”, “projection”.
“_UPPER_FIRST_” Words with lower case letters but capital letter on the first position. Usually person names, locations or organization names. For example: “Larsen”, “Jornada”, etc.
“_ALPHA_”: Words with only alpha characters but not in class “_UPPER_”, “_LOWER_” or “_UPPER_FIRST_”.
“_RARE_”: Words which do not belong to any cases above.

As we see in the figure, the recall rate was increased obviously. Although precision decreased a little bit, we earned a better F-1 score, which shows that we have improved our recall while maintaining the precision of the model. Future works could include more precise model for organizations, since the precision is only 53%.