#CS372 HW03 Report -20150608 Lee, Jun Hyeong

The model extracts features from gap-development.tsv and gap-validation.tsv and trains a Naïve Bayes Classifier with the features and A-coref's. For each snippet, the snippet is tokenized into sentences, which are in turn, tokenized into words, tagged, and parsed. The parser finds noun phrases, which are defined as {<DT|CD|JJ|PRP.+>*<PRP|NN.*>+}. An additional step that automatically re-tags tokens in A and B as NNP (noun proper) was considered, but names with special characters such as 'George H.W.' made it too complicated, so this step was abandoned. We still expect to have all A's and B's correctly tagged as NNP's, as they are human names, and are hardly likely to ever be mis-tagged as anything else.

With the parsed sentences, the following features are extracted as training data.

- pronoun

  The Pronoun itself, which in itself represents its gender and part-of-speech form.

- ratio_distance

  From the parsed sentences, the 'distance' between A and Pronoun is measured as the number of tokens and NP chunks between them. The distance between B and Pronoun is also calculated. Phrases such as 'her sister Kelly' could result in 0 distance as both 'her' and 'Kelly' are in the same NP chunk. Therefore, when calculating the ratio, a small offset of 0.1 is added to both distances. Thus, ratio_distance = (distance(A, Pronoun) + 0.1) / (distance(B, Pronoun) + 0.1).

  This feature is conceptually meant to show how close each words and Pronoun are. A more proper way to get the 'distance' between words would be to parse the text with a context-based approach, such as dependency parsing. But this is a very difficult job as the English language allows many forms of sentences. Therefore we take the above approach, which is much more naïve, but also much easier.

- tag_pre_A, tag_post_A

  These are the pos tags of the token/chunks that come before and after A. If A is at the start or the end of a sentence, the pre and post tags are respectively marked as 'N/A'. These are included to somewhat supplement ratio_distance in extracting

the contextual information of the text.

- tag_pre_B, tag_post_B

  These are the pos tag of the token/chunks that come before and after B.

When usage of the full text of the Wikipedia page is allowed, the following additional feature is extracted:

- ratio_occ

  In the entire text, the occurrences of A and B are counted. If the name is a multi-word phrase, such as 'Linkin Park', it could also be referred to as 'Linkin' or 'Park'. Therefore, the first word of the name, the last word, and the entire name is searched separately. Then, we add the first two and subtracts the latter. The occurrence for 'George W. Bush' is: (number of 'George') + (number of 'Bush') – (number of 'George W. Bush'). The ratio between the occurrences of A and B is ratio_occ.

  If a certain name has more occurrences than another, we can assume that the former is a more importance figure than the latter in the text. This feature is included to task the model to identify any relevance between A-coref and the ratio of such 'importance' of A and B.

After training with these features and the Boolean value A-coref, the model is ready for predictions. Testing with gap-test.tsv yielded the following results:

```
PS D:\학과4\gap-coreference> python gap_scorer.py --gold_tsv='gap-test.tsv' --system_tsv='CS372_HW5_snippet_output_20150608.tsv'
Overall recall: 71.7 precision: 63.6 f1: 67.4
                tp 1272 fp 728
                fn 501  tn 1499
Masculine recall: 71.7 precision: 63.7 f1: 67.4
                tp 637  fp 363
                fn 252  tn 748
Feminine recall: 71.8 precision: 63.5 f1: 67.4
                tp 635  fp 365
                fn 249  tn 751
Bias (F/M): 1.00
```

```
PS D:\학과4\gap-coreference> python gap_scorer.py --gold_tsv='gap-test.tsv' --system_tsv='CS372_HW5_page_output_20150608.tsv'
Overall recall: 73.0 precision: 64.7 f1: 68.6
                tp 1294 fp 706
                fn 479  tn 1521
Masculine recall: 72.9 precision: 64.8 f1: 68.6
                tp 648  fp 352
                fn 241  tn 759
Feminine recall: 73.1 precision: 64.6 f1: 68.6
                tp 646  fp 354
                fn 238  tn 762
Bias (F/M): 1.00
```

The precision rate for snippet analysis is 63.6%, which is better than the 'select True/False randomly' method which has the expected precision rate of 50%. Analyzing the page enhances the precision rate by 1.1%.