Methods

Slightly different approaches were employed for snippet context and page context cases. The initial part is common between the two cases. I first calculated the positions of the pronoun, term A and term B in terms of number of words. I called this 'indices'. In other words, the given offset data was in terms of characters, but I converted them to be in terms of words. This way, it was easier and more accurate when calculating the distance between words later. Then I calculated the positions of the words within the sentence they belong to. I called this 'sentence-indices'. These numbers were calculated because I wanted to ignore words that were in the sentences after that in which the pronoun is located. After calculating the indices, I simplified the given text so that it contains only the relevant information. I removed sentences that does not contain the pronoun, A, or B. The reason behind this was that if a sentence does not contain the pronoun nor the entities, it is not needed to determine which of the entities the pronoun is referring to. Then I recalculated the sentence-indices to account for the sentences that were removed. After this step, the calculations for snippet context and page context cases diverged.

In the snippet context case, I aimed to utilize as much information as possible from the given data. I extracted all possible candidates for the entity the pronoun may refer to. In English, a pronoun may refer to an entity contained in the previous sentence, or an entity that appears before itself inside the same sentence, or an entity that appears after itself but appears in the same sentence. Thus, I need not consider proper nouns contained in the next sentence. To implement this, first I parsed the simplified sentence into a tree structure. Then, I extracted all proper nouns from it. Meanwhile, the gender of the pronoun and the entity should match. Thus, I could ignore proper nouns that have a different gender from the pronoun. To determine the gender of a proper noun, I visited its Wikipedia page. By visiting the page, I could determine if it is a person and its gender. Each entity was labeled 'REAL', 'NAME', or 'NOT' if it is a real person, a name of an imaginary person, or not even a person, respectively. The gender is labeled 'M', 'F', or 'UNK', if it is male, female, or could not be determined, respectively. The position, gender and name of the proper noun is saved for later use. After all the valid candidates are found, I had to determine which of A or B is the correct entity. In 90% of the cases only one of A or B was true. Thus, I used the distances between the candidates and the pronoun to find the one that is more likely to be the correct entity. In the ideal case one of A or B should be the closest candidate, but if not I chose the closer one as the correct one.

In the page context case, I had more information at hand to utilize. I could look at the whole Wikipedia page to extract candidates from it. Unlike the snippet context case, I did not use the RegexpParser to extract proper nouns. Instead, I collected terms that had a Wikipedia page of its own, hypothesizing that entities that the pronoun refers to is likely a Wikipedia entry. Only the entities that appear before the pronoun is counted, just like the snippet case. Also, the title of the page was added to the end to give an advantage to it. Then, the entity type and gender was labeled as I did in the snippet context case. Choosing from the candidates was similar to the snippet context case. Because I assumed the correct entity was one of Wikipedia entries, I checked if either A or B is contained in the candidates. If both were not one of candidates, they were both labeled false. If both were not the best candidates, the closer one was chosen. By design it is impossible that they have same distances.

Discussion

The results of the snippet context case and the page context case are shown in Figures 1 and 2.

The performance of the page context case was better than the snippet context case, as expected. This was because there was more information available to decide the coreference through the Wikipedia page. Checking the test data, most of the entities for A and B were Wikipedia entries. This is natural because the author of an entry links the important term to the corresponding page.

```
Overall recall: 31.7 precision: 40.4 f1: 35.5
                tp 562  fp 830
                fn 1211 tn 1397
Masculine recall: 33.0 precision: 40.8 f1: 36.5
                tp 293  fp 425
                fn 596  tn 686
Feminine recall: 30.4 precision: 39.9 f1: 34.5
                tp 269  fp 405
                fn 615  tn 711
Bias (F/M): 0.95
```

Figure 1 Snippet Context

Also, most of the URLs provided were pages about a person, either real or not real. It is natural to conclude that text in an encyclopedia page describes the entry. Thus, giving an advantage to the title of the page, so that A or B that is the title gets true as the label, was a good idea.

```
Overall recall: 39.0 precision: 67.8 f1: 49.6
                tp 692  fp 328
                fn 1081 tn 1899
Masculine recall: 40.6 precision: 65.5 f1: 50.1
                tp 361  fp 190
                fn 528  tn 921
Feminine recall: 37.4 precision: 70.6 f1: 48.9
                tp 331  fp 138
                fn 553  tn 978
Bias (F/M): 0.98
```

Figure 2 Page Context

In addition, the provided data had only male or female pronouns. This lead me to think of ways to exclude candidates based on gender. There were lists of male and female names available online, but it was not comprehensive enough to cover the test cases. Thus, I additionally checked the Wikipedia page of the name to see which of 'he' or 'she' appears in the text. This gave some improvement in accuracy because a small list of names lead to many entities being labled as 'unknown gender', making it difficult to choose from the candidates.

Furthermore, the labels for A and B coreferences were mostly true-false pairs; there were not many true-true or false-false pairs. This simplified the problem into deciding which of the two entities is more likely to be the correct one. The main criteria for deciding the correctness was the distance because pronouns refer to the close entity rather than the one mentioned long time ago. In the snippet context case, the distance was considered rather naively, but in the page context case, the process became more sophisticated by considering only those that are Wikipedia entries.

Possible Improvements

The title of the Wikipedia page was added at the end as a candidate, favoring it as the correct entity. However, it could have been too much of a favor in some cases, if the pronoun did not refer to the title of the page. I thought of giving a slightly higher score to the title instead, but the amount of score was too arbitrary. If there was a systematic way of favoring the title, it could improve the accuracy in the page context case.

When checking the Wikipedia pages in page context cases, there were some difficulties in locating the position of the provided text. I realized this was because the page was updated after the dataset was compiled. To account for such cases, I had to use a smaller substring of the text to locate it. This might have caused some inaccuracy in the position of the text if there were multiple occurrences of the substring in the page. The accuracy would have been more accurately reflected if the dataset and Wikipedia were synchronized.

Also, special alphabets such as é and ü were replaced with an asterisk in the dataset. There were cases in which it confused the RegexpParser, resulting in inaccurate candidates, and the original character is unknown so it could not be searched on Wikipedia. If such characters were kept, it would have helped in both snippet context and page context cases.