Data Description:

The original Enron email data set are saved in plain text file without format, such as html, MIME, or xml. For further experiments purpose, We develop a parser that convert all this email into json format. Furthermore, all threads inside an email body are also extracted. The following numbers describe the stats of the output json data.

```
Number of threads: 170507
Number of emails with thread: 450339
Number of non-thread emails: 346894
Total files: 517401
Total emails: 797233
```

Distribution of threads in email:

thread	1	2	3	4	5	6	7	8	9	10	11+
number of emails	346894	113759	34277	12133	4804	2210	1189	600	403	246	874

Mention detection:

Mention detection is run by `nlp4j`. The performance is measured by the precision, recall and F1 score. These score are calculated by 11 threads which include 30 emails. All these emails are tokenized into 476 tokens. Totally 78 mentions are annotated, but 3 mentions are false. There are 13 mentions that `nlp4j` miss.

F1 score

```
precision: 0.9615
recall: 0.8523
F1 score: 0.9036
```

The reason why the score of recall is relatively low is because the performance of annotating organization can be improved in `nlp4j`. However, the purpose of this project is to find relationship of people inside email data set. Organization appearing in email usually refer to nothing. Thus, we can proceed to next step by ignoring the effect of organization.

Amazon mechanical turk annotation:

There are 43 threads and totally 116 emails picked and deployed to amazon mechanical turk.

For each annotation, we ask two people to annotate to ensure the agreement. The Bcubed score are used for evaluating the quality of annotation.

Bcubed Score

```
precision: 0.9566
recall: 0.9553
fscore: 0.9518
```