**Purpose**

Design an aggregation method

Dataset:

Teams->files->class materials->homework 3

1-7.tsv and chemprot\_test\_entities.tsv has the same format: (paperID, label, start\_offset, end\_offset, text\_span)

chemprot\_test\_abstracts.tsv contains the abstracts of the papers.

**Submission**

The following files in a zip file. No report for this homework

1. aggregation.py
2. evaluation.py
3. readme.txt

How we will test your algorithm

1. We use python 3 by default. Numpy, Scipy, Spacy are installed. If you need a different environment or additional packages, please put detailed instructions in the readme file.
2. We will put the aggregation.py in the same directory with the following files:
   1. 1.tsv, 2.tsv, …, 7.tsv  
      note: we will shuffle the 1-7.tsv files (eg. 5.tsv may become 7.tsv in our testing)
   2. chemprot\_test\_abstracts.tsv  
      note: chemprot\_test\_entities.tsv is not in this directory
3. we will run your aggregation code as “python aggregation.py”
4. we will run it for maximum 90 minutes on a computer with 16G RAM, quad intel core i7, no GPU.
5. Your aggregation.py should write the aggregation output as output.tsv, same format as chemprot\_test\_entities.tsv.
6. For evaluation, we will put evaluation.py, output.tsv, and chemprot\_test\_entities.tsv into the same directory.
7. We will run your evaluation.py as “python evaluation.py”. You need to implement span level precision, recall, and F1. Use print("precision: %f \n recall: %f \n F1: %f" % (pre,rec,f1)) to print the results.

### Grading

Your homework will be graded by the performance of your algorithm (span level precision, recall, and F1, and efficiency). The top 5 algorithms will get bonus points. Please make sure your code is executable. We will do our best to debug if we encounter error messages, but you may get deductions. If we have a problem running it, you will be contacted.

### Deadline

Nov 15 11:59pm

Late submission: 25% reduction for each day