

# CSE 4334/5334 – Data Mining

---

## *Fall 2014 - Project 2*

*Due: 11:59pm Central Time, Tuesday, Nov 11, 2014*

### **REQUIREMENTS:**

Read the following requirements carefully, and make sure you follow every rule. If you fail to meet some requirement, marks will be deducted accordingly.

- Submit **ONE zip file** that contains all source codes, 3<sup>rd</sup> party libraries and a single project document. **(5 points)**
- Requirement on your project document:
  - No limitation on the number of pages, or the format of the document;
  - The document should clearly describe how you design and implement your ideas; **(5 points)**
  - The document should clearly state how to execute your program; **(5 points)**
  - The document should provide execution screenshots of your program. **(3 points)**
- Requirement on your source code:
  - You have to implement the whole project by yourself;
  - 3<sup>rd</sup> party libraries can be used **ONLY** for reading TSV files;
  - Your source code must pass compilation. Any non-executable submission is not acceptable. **Make sure your program compiles and runs on omega.uta.edu, before you submit.**
  - Your submission must EXCLUDE the input files that we provide to you; **(2 points)**
  - You can use **any language that can run on omega.uta.edu**, though I recommend Java and Python;
    - If you use Java, you may include your Eclipse or NetBean project folder. Remember to exclude the input files;
    - If you use python, then you only need to submit your source code.

### **PROBLEM SCENARIO:**

#### **BACKGROUND**

Project 2 asks you to predict which jobs users will apply to. This will give us a basis for recommending jobs to career website users. For satisfactory user experience, it is important to only recommend jobs that interest users.

#### **TASKS**

You are given 5 files:

- *jobs.tsv*: The same file used in Project 1;
- *users.tsv*, *apps.tsv*, *user\_history.tsv*: The schema and format of these files are identical to that of the files used in Project 1, but they now contain more users and their applications/history

information;

- *user2.tsv*: A single-column file that contains the UserIDs of a subset of the users in *users.tsv*.

Conceptually, users are partitioned into 2 mutually-exclusive sets---those in *user2.tsv* (denoted by U2) and those not (denoted by U1). Timestamps are partitioned into 2 ranges---before 2012-04-09 00:00:00 (denoted by T1) and after 2012-04-09 00:00:00 (denoted by T2). The file *apps.tsv* contains all those applications made by U1 (during both T1 and T2) and all those applications made during times in T1 (by both U1 and U2).

Your task is to predict what are the jobs U2 have applied to during T2. (One thing to remember is that no one can possibly apply to a job during T2 if that job's EndDate is before 2012-04-09 00:00:00.)

More specific tasks include:

- 1) **(10 points)** read information from input files.
- 2) **(50 points)** build your prediction tool.
- 3) **(20 points)** print the prediction results to an output file named *output.tsv*. It should look like the following. An example file *sampleoutput.tsv* is given to you.

```
1471976      1020868
1471976      628097
... ..
1471976      284009
1471983      628097
1471983      891097
...
```

Your output file *output.tsv* should contain 150 lines, each of which has two tab-separated fields (UserID, JobID). The UserID must belong to U2, and the JobID must have an EndDate after 2012-04-09 00:00:00.

The 150 pairs of (UserID, JobID) should be ordered by how likely UserID has applied to JobID during T2. (It is known that a UserID doesn't apply to the same job twice. So if you find an application about UserID and JobID in *apps.tsv*, the pair shouldn't appear in *output.tsv*.)

We will use your *output.tsv* to assess how accurate your prediction is, including whether more likely applications are ordered before less likely ones. (We have ground truth data about all the job applications made by U2 during T2.)

To accomplish the tasks, you need to look for clues from users' previous applications, demographic information, and work history. You should consider compare different approaches and tune and improve your prediction.

Your program should be executed by the following commands:

```
java your-main-class-name /path/to/data/file/directory/ /path/to/output.tsv
or
python your-script-file.py /path/to/data/file/directory/ /path/to/output.tsv
```

or

```
./a.out /path/to/data/file/directory/ /path/to/output.tsv
```

\* */path/to/data/file/directory/* is the path (e.g., */home/john/data-mining/data/*) to the directory that has all 5 input files: *users.tsv*, *jobs.tsv*, *apps.tsv*, *users\_history.tsv*, *user2.tsv*.

\* */path/to/output.tsv* is the path to the output file, e.g. */home/john/data-mining/output.tsv*