Alex Scharf
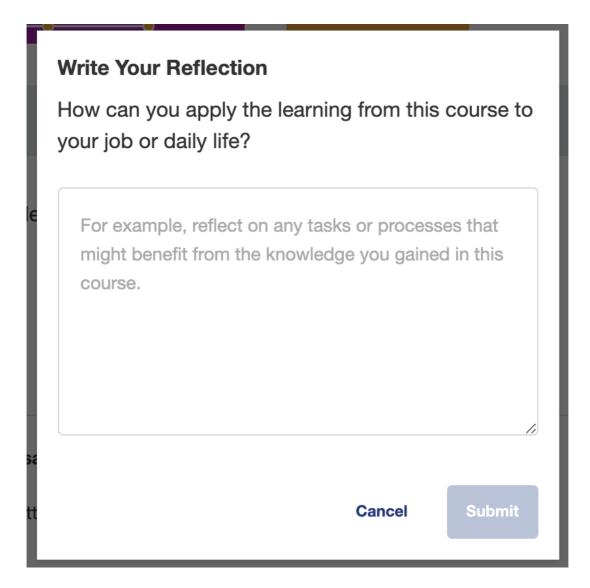**Project Proposal, CS410**
Free Topic - English and Japanese Course Reflection Analysis

1. What are the names and NetIDs of all your team members? Who is the captain? The captain will have more administrative duties than team members.

Individual project, Alex Scharf (acscharf2)

2. What is your free topic? Please give a detailed description. What is the task? Why is it important or interesting? What is your planned approach? What tools, systems or datasets are involved? What is the expected outcome? How are you going to evaluate your work?

I am the product owner of GLOBIS Unlimited (GLOBIS Manabihodai) in Japanese, an online course platform focused on business content with currently over 200,000 learners. After learners complete a course on the platform, they are prompted to enter an optional reflection about how they can apply the learning from the course to their job or life:

Write Your Reflection

How can you apply the learning from this course to your job or daily life?

> For example, reflect on any tasks or processes that might benefit from the knowledge you gained in this course.

Cancel   Submit

Some of the responses are quite good and well-thought out ("I am an engineer, and I can use logic trees to help me create test cases") and others are not as insightful ("Logic trees are good"). After learners post their reflections, they can view the reflections of other learners and "Like" ones they found useful. As learners see the reflections of others, it is important that the reflections be of high quality, and hence useful to a wider audience.

Because the service is in English and Japanese, data exists for both Japanese and English reflections.

I would like to do the following:
A.) Identify language and parts of speech that are common in reflections for both English and Japanese
B.) Create an application that predicts whether an input reflection is "high quality" or "low quality" for English only
C.) Host that application on a public web server

This task is important and interesting for the following reasons:
- Real-world challenge based upon actual data set
- Creates a proof of concept for a feature that would give learners feedback about the quality of their course reflection, leading to great personalization in online learning
- Creating a user-facing application that potential users could interact with
- Working across two language (Japanese and English) that are very different

My current plan is as follows:
1.) Extract data set for both English and Japanese
2.) Choose the most appropriate way to score the data set (manual approach or use "likes")
3.) Find tool that can analyze and make predictions for both English and Japanese
4.) Run analysis of common words in reflections to satisfy point A
5.) Train predictive model on current data set
6.) Create application that allows user input and compares against predictive model
7.) Host application on a publicly accessible web server
8.) Run user test with 4 people to see if their experience is in line with expectations

I expect that the application should be able to identify "good" and "bad" reflections to a certain degree. I plan on evaluating my work if the test users enter a "good" reflection and a "bad" reflection and the application is in line with their expectations.

Here is what I imagine the application flow to look like:

1.) User is prompted for reflection input and submits
2.) System checks user input against trained data and decides whether it is "high quality" or "low quality"
3.) Feedback is sent back to the user

3. Which programming language do you plan to use?

I plan on using Python

4. Please justify that the workload of your topic is at least 20*N

hours, N being the total number of students in your team. You may list the main tasks to be completed, and the estimated time cost for each task.

- Research tool that is appropriate for both Japanese and English (2 hours)
- Extract data set and prepare in proper format (3 hours)
- Score the data set in appropriate format (3 hours)
- Create application to run semantic analysis of reflections (5 hours)
- Create application to train predictive model (5 hours)
- Create front-end application that allows users to input data against predictive model (5 hours)
- Host front-end application on publicly accessible web server (2 hours)
- Prepare for user test (1 hour)
- 4 user tests x 30 minutes (2 hours)
- Create full documentation (2 hours)
- Summarize findings (2 hours)
Total: 32 hours