Instructions for Submitting Final Code

Deadline: Sunday May 8, 2022 11:59 PM.
We will be starting our GCP instance for final submissions on Monday 05/02/2022.

Please go through the following instructions carefully to get an idea of what you need to submit:

Code Requirements

1. The code should not contain any absolute paths or custom paths. To ensure this, you might have to run some find-and-replace commands/scripts. For example, your wav.scp file might contain absolute paths. Make sure you run your inference (decoding) script on our server to make sure it works in that environment.

2. If you worked in the kaldi environment, only transfer your modified egs. If your project is based on one or more Kaldi egs recipes, you need to provide a diff of your code against the kaldi code. Please use the `diff -ru` command to do that. Here is an example from the professor, assuming that the code was derived from egs/voxceleb/v2. Let's say the modified version is egs/voxceleb/v2_beigi. Then do the following:

In the kaldi directory, form the Linux command do the following:

```
diff -ru egs/voxceleb/v2 egs/voxceleb/v2_beigi > egs_voxceleb_v2_beigi.diff
```

Now you should submit `egs_voxceleb_v2_beigi.diff` in the main directory of your code sumbission. Of course you would replace beigi with your Uni and use the proper egs.

3. We will be running the inference (decoding) code on your models on our VM. This should not use any GPUs. For this, you should upload your trained models.

4. We should be able to see all your code and be able to run your code through the main script without making any changes and without any need to pass any arguments. All arguments should be defined inside the script. Run your code through the script to make sure it works properly.

5. For inference, you should ensure that the test split of your processed data is present and accessible in your script path so that it would run without errors and would report the relevant results.

6. If you used a non-standard dataset, you need to upload the whole raw dataset.

7. Decoding should not take more than 15 minutes.

8. A clear README with instructions to run the script should be provided. More instructions about the README:

Submit a README.txt file which is in ascii text. Please do not provide .docx, .odt, etc. Only pure text. The README.txt file should be very comprehensive and have the following sections:

a. UNI and Full Name

b. Date

c. Project Title

d. Project summary (abstract -- one paragraph)

e. List of tools that were used and may possibly be needed to run the code. Also, include URLs to any such tools if they are anything that was not included in Kaldi.

f. List of directories and executables that may be used to test the code.

g. The name(s) of one or two main scripts that run everything. For example (run.sh or a train.sh and test.sh, etc) with a short description of what they do and a syntax of how to run them. You should include a single line example of how to run a sample with all the arguments necessary to run a sample. Although you show the syntax, you should also show an example with actual filenames that need to be passed, if any. Also, provide the sample output and where to look for it, if we run your example.

h. This file should also include any references to where the data was obtained, what opensource code is being used, etc. by providing explicit references including links if needed.

Please try to stick by the above minimal information. You can also provide any other extra information that is necessary to understand and/or run your code. The easier you make the code review, the higher chance that you get a good grade. You may have done a wonderful job, however, if you do not do a good job in the README.txt file, we may never see the merit in your code.

9. A scripts directory that contains all your scripts. The order of calling these scripts with detailed information about the arguments that need to be passed to each script should be included. Whenever possible, you should have a main script which runs through everything with no intervention needed.

10. Each piece of code that you have written, whether in a new file or modification to existing files should be commented in detail with your UNI specifying you as the author of that portion. This way we can search for your contribution by searching for your UNI. Also, the README.txt file should explicitly specify which files you have touched. The comments should include pseudo-code of what the function the piece of code is trying to achieve, including the input and output.

11. If you are using python, create a virtual environment in your directory on our instance and install all dependencies there. Mention how to activate your virtual environment in your README.

Note: If you are still facing issues while SSHing into your VM, please email sr3846@columbia.edu directly, and we will look into your individual case.