

CS 6320.002: Natural Language Processing

Fall 2019

Project Milestone 3 – 50 Points

Due 8:30am 20 Nov. 2019

Deliverables: A tarball containing the following (submit just one per team):

- A PDF writeup. Put the names of ALL team members on your writeup.
- Python files implementing your chosen baseline system.
- A trained baseline model, `baseline.model`.
- A Python file `evaluate.py`.
- The test set for your project.

0 Planning Ahead

After Milestone 3, you have the following milestones:

- Milestone 4, due 2.5 weeks after Milestone 3. Implement your improvements over the baseline and evaluate. Complete your final project paper.
- Presentation, also 2.5 weeks after Milestone 3.

Each milestone is worth 8% of your final grade; the proposal and presentation are worth 4% each.

You don't have to (and shouldn't!) wait for the milestone due dates to work on each stage of your project! Keep working steadily so that you aren't rushing at the end of the semester. There are no more homeworks in this class, so set aside the time you would have spent each week doing homework to work on your project instead.

1 Baseline System – 20 points

For this milestone, you must reimplement a published baseline system for your task – probably from one of the papers you summarized in Milestone 1, but you can choose a different baseline system if you find a new paper.

Writeup Question 1.1: Write 1-2 paragraphs describing your chosen baseline. What paper was it published in? What machine learning algorithm does it use? What kind of preprocessing does it require? What are the features?

You can use as many files as you want to organize the code for your baseline system. For your implementation, you may use machine learning or NLP toolkits (eg. SciKit-Learn, Spacy, NLTK) to for preprocessing and models, but you must code the rest of the system from scratch. You may NOT submit existing code that implements the system (eg. that you found on a GitHub repo).

Writeup Question 1.2: Give step-by-step instructions for how to run your baseline

system. What libraries are needed? What commands are used to run it, with what arguments, in what order?

2 Trained Model – 10 points

Train your baseline system and save the resulting model in a file named `baseline.model`. For example, SciKit-Learn models can be saved using the `pickle` module, and Pytorch models can be saved using `torch.save`, which uses `pickle`. You do not need to submit the training data, only the trained model.

Writeup Question 2.1: Write 1-2 paragraphs describing the training process. What learning scheme did you use (unsupervised, semi-supervised, or supervised)? Were there any hyperparameters, and if so, how did you tune them? Did you run on a CPU or GPU? How long did it take?

3 Evaluation – 20 points

Write a file `evaluate.py` that does the following:

- Load your trained baseline model from file.
- Load the test data.
- Make predictions/generations for the test data using your trained model.
- Evaluate the predictions/generations using an appropriate metric for your task.
- Print the metric scores.

Writeup Question 3.1: Give step-by-step instructions for how to run `evaluate.py`. What arguments does it take?

Writeup Question 3.2: Write 1-2 paragraphs describing the evaluation. What was the performance of your baseline system? Is that good or bad? Why is the metric appropriate for your task? If there are other metrics commonly used for the task, why did you choose this one?