

Overview

For the final project, students will work in teams of 1 or 2 to implement a novel system or analyze an existing model using a dataset they deem appropriate.

Part 0 (10%). Decide on team / topic.

Part 1 (30%). Implement a baseline (optionally collect a dataset if one does not exist). Put together a writeup with background on the topic of interest.

Part 2 (30%). Implement a more complex system (either to beat your baseline or for a separate but related task).

Part 3 (30%). Final write-up. Make any code / writing revisions that were previously recommended.

By the end of the project the students should have a github repo with their code, a clear and presentable writeup, and strong background knowledge in their chosen topic.

A list of project ideas are available here: <http://bit.ly/cs5304-project-ideas>

Please refer to the course site for information on deadlines: <https://cs5304.github.io/>

The 3-slip day policy does **not** apply to project deadlines.

Part 0: Project Proposal

Submit a writeup on CMS (1 per team) with the following information:

- A. The names and netids of the students in your group.
- B. Project title (you can change this later).
- C. A project proposal (no more than 2 paragraphs, 3-5 sentences each) that clearly but briefly describes a machine learning problem. It should mention whether there is available data (your intent to collect data if necessary), an approach that you may take, and what you may expect to find. You should also include three research papers that you plan to use as reference.

Example:

Generating Movie Ratings from Unlabeled Text Data

Movie ratings have become prevalent thanks to the ease of access of viewing films (through streaming services like Netflix) and standalone platforms for movie discovery (such as Rotten Tomatoes). In some cases there is an explicit association with a written review and its sentiment value (represented as integer 1 through 10), although the

internet is vast and contains many written reviews without this type of labeling.

For our project, we are interested in training a sentiment classifier of Rotten Tomatoes data (SST), scraping movie reviews from the subreddit r/netflixbestof, and examining whether we can use these data and model to generating ratings for movies. We'll experiment with both bag of words and n-grams classifiers implemented as neural network models. We'll use pre-trained word embeddings and will need to write some preprocessing code for associating unlabeled text reviews with specific movies.

[1]: Socher et al. (2013) Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank

[2]: Joulin et al. (2016) Bag of Tricks for Efficient Text Classification

[3]: Yoon Kim (2014) Convolutional Neural Networks for Sentence Classification

D. Two shorter alternate proposals (no more than 3 sentences each).

Part 1: Implement Baseline

The objective of Part 1 is to implement a machine learning model that serves as a baseline for the dataset and task that your team proposed in Part 0.

You will submit a write-up that should contain a link to your code, a description of your model, and the results. In order for the reader to appreciate why this is a fair baseline, it is necessary that you provide a significant introduction, provide descriptions of related work, explain the dataset, and explain your experimental setup.

We do not yet expect you to explain the future model that you will implement for Part 2, nor do we expect any serious conclusion besides observations and analysis that your baseline has been reasonably chosen.

You should aim for somewhere between 3 to 4 pages, no more than 5. As a reference, for the final write-up, the guideline will be 4 to 5 pages, no more than 6.

Here is an example layout for the writeup:

Page 1: Abstract, Introduction, Related Work

Page 2: Baseline Model Description, Dataset Description

Page 3: Experimental Setup, Results and any Analysis, Citations

Page 4: Citations (cont.)

Part 2 and 3: Introduce Your Model and Analysis

The objective of Part 2 is to build upon the baseline using a modified or new model. You should provide analysis that compares the different models and a conclusion summarizing what your findings are.

The structure of the writeup will likely be the same as for Part 1. The guideline for number of pages is also the same: 3 to 4 pages, no more than 5.

By this point, your project should have introduced some new experiments / results / analysis that were not included within the related work section of your paper. This is an impressive feat!

For Part 3, we ask that you go even further. Run more experiments and alternative analysis. Perhaps put effort into some data visualizations. The outcome should be a more polished and more comprehensive project than you had accomplished for Part 2.

For Part 3, you'll also be required to give a 3 minute presentation about your project during class (and field an additional 1 minute of questions from the audience). There are 40 projects, so it's important that we stay on schedule. We'll give cues when your time is running out while presenting, although you should really practice and attempt to go for less than 3 minutes if possible.

Your grade for Part 2 will be based on how compelling of a story you can tell with your model, experiments, and analysis. The grade for Part 3 will be based on the comprehensiveness of your final write-up and the clarity of your presentation.

Note: All experiments and results should be reproducible. This is easiest done by providing a link in your writeup to a github repo containing your code with some basic documentation.

Looking Forward

The project will involve a major programming portion (implementing and training the models, including any feature engineering or other preprocessing steps), a major research portion (reading papers to understand your problem deeper and to get a sense of findings from previous work), and a major writing portion. The final writing portion will also need to include section explaining the responsibilities of each student in the team throughout the project. We'll give slightly more detail for the expectations of each part of the project as the timeline progresses.

A team with 2 people should be doing roughly 1.5 times the amount of work as a group with one person. For this reason, it is probably beneficial to work in a group if possible. That being said, if 1 person in a 2 person team does not carry their weight, then it's possible the project will fall below the expected baseline.

For the most part, we will require that each group works on completely distinct projects. We'll try our best so that each group gets their first choice, but may ask people to slightly modify their project based on the interests that they've indicated in their alternate proposals. In the

case of two groups having nearly identical project proposals, the group that has submitted first would get preference.

Writeups (except for part 0) must be in LaTeX using the ACL 2018 format. Template available on overleaf: <http://bit.ly/cs5304spring2018-project-writeup-template>