**CS 485 / CS 584 Text Processing with Neural Networks** **Spring 2024**

# Homework 2

*Lecturer: Dr. Fei Liu* *Due: Wednesday, 4/3 11:59PM EST*

**Note**: *Adapted from homework by Eric Xing at Carnegie Mellon.*

## 2.1 Text Classification (15 points)

For this question, you'll be using the 20 Newsgroups dataset for a binary text classification task. You can download the dataset from `http://qwone.com/~jason/20Newsgroups/` and choose the file named **"20news-bydate.tar.gz"**. Unpack it and look through the directories at some of the files. The documents are divided into a training set and a test set. Overall, there are 20 categories and roughly 19,000 documents. The label (category) of a document is its folder name. For this homework, you will only be using **two categories**: `rec.autos` and `rec.motorcycles` in the train/test folders.

Your task is to implement a naïve Bayes classifier for text classification. First, you may want to convert the collection of documents into a matrix of token counts. See `https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html` and `https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_20newsgroups_vectorized.html`.

To build the classifer, you can modify the provided starter code in HW1 or implement your own naïve Bayes classifier using an existing library, such as `sklearn`. (See `https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html`). If you choose to modify the starter code, be sure to modify the train/test splits so that the classifier will report results on the provided test set.

- **(10 points)** Evaluate performance of your text classifier on the test set (i.e., documents in `rec.autos` and `rec.motorcycles` folders) by calculating the **F1-score**, assuming `rec.autos` is the positive category (see `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html`). What scores do you get?
- **(5 points)** Choose four documents incorrectly classified by your Naive Bayes classifier. Design a simple prompt for classifying these documents. You can find an example in the "Zero-shot Classification" section of this article: `https://cookbook.openai.com/examples/multiclass_classification_for_transactions`. Next, test your prompt with both OpenAI's ChatGPT (`https://chat.openai.com/`) and Anthropic's Claude (`https://claude.ai/`) by creating accounts on their platforms. Document the prompt, each test document, and the labels as predicted by Naive Bayes, ChatGPT, and Claude. Do LLM models offer better accuracy? Also, do ChatGPT's and Claude's predictions agree?

**Please submit**: (1) A report named `report_firstname_lastname.pdf`. In your report, answer the above questions and describe your experimental setup. Include information such as the programming language used, running time, the prompt, test cases, the LLMs used, and any other relevant details. (2) Source code of your implementation in a zipped or tar file.