NLU CSCI 4907/6515 - Spring 2026
Project guidelines

The goal of the final project is to deepen your familiarity and get hands-on experience with the field of NLU/NLP. This will be a team project, where I recommend the size of the team to be 3-4 (but open to having 2).

Deadlines (details below)

March 6th: Project proposal due
April 27th: Final project presentation
May 5th: Final project report and code due, and any other related materials (as applicable to the project)

Potential project types

- Build a model to perform a task: Decide on a task that you would like to try (e.g., sentiment analysis) and apply some of the methods we've discussed in class to this task. Report on how well your model does, what some of the obstacles seem to be, and how you might improve the model moving forward.

- Replication: Choose an existing research paper and try to recreate the results (or a subset of the results) of the paper. Find a peer-reviewed paper that reports specific results on a specific dataset and implement what they describe in the paper. Compare your results against theirs (suitable for smaller team)

- Model probing/analysis/evaluation: Choose an existing research paper or algorithm and analyze its behavior, propose and test a new method of evaluating it, test/analyze the role of a particular parameter, test whether it works on a particular type of data/input, and so forth. For example, you could test whether language models are able to handle a particular linguistic phenomenon, or input type.

Project ideas

This list is only a suggestion to help you, but I would like to see each team come up with a creative idea/plan and discuss them in the proposal. You can come to me with clarifying questions about the topics, for help brainstorming how to approach the experiments, to seek assistance in obtaining data, etc. If you have a different idea for a project that you would like to pursue instead of any of the topics below, you are more than welcome to do so. In all cases you

will be submitted your proposed topics to me and I will provide comments about whether any adjustments are advisable.

- ❖ Language Model Exploration. Implement and train a language model on some training data, and report the perplexity of this model on a held out set of test data. Compare this perplexity against (at least) one of two things: 1) perplexity of the same model on test data from a different domain, or 2) perplexity of a different language model on the first set of test data. Examine some of the probabilities output by the model(s), and discuss why do you think the results come out with the differences and/or similarities that they do. Discuss the implications of your findings.

- ❖ Model analysis / Language models: For this project, take a pre-trained large-scale language model (like BERT / ChatGPT, etc.) and compare its output against how humans behave. For example, when predicting upcoming words, do the models match what humans do, or not? What kinds of differences do you observe? Do you have an explanation for this difference?

- ❖ Model analysis: Does a model of your choosing (e.g., large language models: BERT, GPT, word embeddings) exhibit biases? Do we observe similar biases across languages or corpus types?

- ❖ Replication (team of 2, maximum of 3 but you would need to add more tasks than what I suggested below):
  - ➢ "To BERT or Not To BERT: A Study of Pre-trained Word Embeddings for Clinical NLP"
    - ■ Project: Compare standard BERT vs. Domain-specific embeddings on a public medical dataset (like MIMIC-III subset or simpler biomedical NER tasks).
    - ■ Undergrads run existing models to compare F1 scores. Grads must analyze *why* one fails (error analysis) or fine-tune the "Not BERT" baselines to see if they can beat the heavy model.
  - ➢ "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks" (Gururangan et al., 2020)
    - ■ Project: Pick a domain (e.g., Legal text or Tweets).
    - ■ Undergrad Task: Fine-tune BERT directly (Baseline).Compare pre-training on a huge domain corpus (DAPT) vs. pre-training only on the unlabelled training set of the specific task (TAPT).
    - ■ Grad Task: Implement the "Domain Adaptive Pretraining" (DAPT) phase and measure the lift in performance.

- ❖ Naive Bayes / Logistic Regression for NLP. For this project, implement and train a Naive Bayes or Logistic Regression classifier for some NLP task (you can, for instance, use one of the tasks / datasets reported on in the BERT paper). Your set of features should expand beyond the features illustrated in the Jurafsky & Martin chapter on Naive

Bayes classification. You should also get the results of at least one baseline model, against which you can compare your model's performance on your chosen task. Discuss your results, why you think they turn out the way they do, and their implications. A good way to add substance for this project would be to analyze and discuss the errors made by the model, and/or to submit intentionally challenging ("adversarial") inputs that you think the model will fail on, and discuss the results.

- Neural Network for NLP. Design a neural network model and train it to do an NLP task (for instance, one of the tasks reported on in the BERT paper). You will need to choose what your model architecture will be, and what form the input to the model will take (e.g., full words or characters/partial words? provide a fixed number of inputs at once in a feedforward format, or give inputs one at a time in recurrent format? etc). You should also get the results of at least one baseline model, against which you can compare your model's performance on your chosen task. Discuss your results, why you think they turn out the way they do, and their implications. A good way to add substance for this project would be to analyze and discuss the errors made by the model, and/or to submit intentionally challenging ("adversarial") inputs that you think the model will fail on, and discuss the results.

Tips:

- Take a look at Best Paper Awards at relevant conferences: ACL, COLING, NAACL, EMNLP
- Look on Google Scholar: you can find papers that were cited by a paper you found, or that cite the paper you found
- Look at challenges on Kaggle
- Check out Papers with Code: https://paperswithcode.com/
- The textbook has many references. Take a look back over sections that were interesting to you and look at any papers that strike your interest.
- To the extent possible, let this help “future you” (or present you)! For example, if you’re going to be applying for jobs and need a coding sample, pick a project that will end with that.
- Err on the side of a smaller project!!! I cannot stress this enough. A bite-sized project is totally acceptable. I prefer a coherent, high-quality small project than a large, ambitious, and unfinished one.
- You can change your project as time goes on (I strongly recommend not to do that, but please submit a new proposal if you choose to do so, so I can approve your new idea and provide justification why you are changing. The deadline for changing is March 17th.

Project proposal components (due March 6th):

The purpose of the project proposal is to lay out your project as concretely as possible,

and for me to provide feedback on the feasibility/relevance/appropriateness of the project.
Please include:

- ★ Names of team members
- ★ The questions your project will address
- ★ Brief (~1 paragraph), but CONCRETE description of what you're planning to do for your final project.
- ★ A breakdown of who will do what.
- ★ If you're doing a replication, provide the paper's reference and abstract, a summary of the paper, and the methods/results you will be replicating.
- ★ If you're doing a model analysis, provide the paper's reference and abstract, a summary of the paper, as well as a link to the code/dataset for reproducing the results. In addition,
- ★ please provide what you will be analyzing (e.g. I want to test whether this neural network model "understands" the notion of negation).

Project write-up components (due May 5):

The length and format of your paper will depend entirely on the project you complete. For most projects, I expect something like 4-8 pages will be sufficient for you to describe in detail what you're doing and what you learned. Your final paper should not exceed 8 pages in length. If you feel like you need more room, please come talk to me.
The format of the paper will depend on your proposed topic, but may include:

- Introduction: sets up the problem you're addressing, why it matters (often contextualized briefly with respect to the relevant literature/course material), how you plan to approach it, and a brief summary of what you found
- A Model or Methods section describing the model/methodology that you're using
- An Experiments section describing additional details about how you ran your
- experiments.
- Results section reporting the results obtained in your experiments
- Analysis section: You may include an additional analysis section for follow-up analyses/experiments that you run. For some papers, this may be the central component of your project. For some projects this may just be a good way to add more substance/arrive at clearer conclusions/ensure that there is enough work for all members.
- Discussion: section discussing the results and their implications with respect to the problem you set out to address.
- Conclusion / Future Work: state your general conclusions. It's also good to discuss logical next steps that could be carried out in future work.
- Ethical statement: What are the broader societal and ethical implications of this work, if/as relevant?
- References: Include a proper citation and bibliography entry for each paper you reference in your write-up. You should also cite any work referenced in the write-up, even if it's not one of your core reviewed piece.

Grading:

- [ ] The project proposal will be graded pass/fail. I will provide feedback on feasibility and relevance, as well as any other suggestions I have. If you're unsure whether your project idea works, please email me as soon as possible.
- [ ] The final paper will be graded on completeness/adherence to your project plan, quality of the work, understanding of the topics, clarity of the writing, coherence of your questions, methods, analyses, conclusions, etc., as applicable.
- [ ] Do you lay out a clear, well-motivated question/problem to be addressed? Do you choose methods that are well-suited to addressing that question/problem, and justify why those methods are appropriate? Do you report your results clearly, and include sufficient, appropriate analyses to allow for substantive takeaways? Are your interpretations of the results/analyses coherent, clear, and logical? Do you clearly lay out the conclusions that your results imply, with respect to your original question/problem?
- [ ] All code should be well-commented, runnable, and living on github
- [ ] No large language models/paraphrasers should be used to write anything that will be handed in the final report. This will result in a 0. If you use LLM for part of your code, please add a comment next to the code section and this should not be more than 15% of your code.