9.190 Class Project Guidelines
Due date: 8 December 2023

October 2, 2023

1 Introduction

Students enrolled in the graduate version of Computational Psycholinguistics (9.190, as opposed to 9.19) are required to complete a class project. This project is due **December 3**. This project can take any of a number of forms and can be on any of a number of topics, but it should have some sort of computational modeling component (understood broadly), so that the project builds on the learning and assignments work you've been doing in this subject during the semester. We also strongly encourage coupling your computational modeling work with empirical analysis of some linguistic dataset and/or with behavioral experiments (e.g., through Mechanical Turk). We encourage you to choose projects that are synergistic with your current or planned research. There may also be opportunities to work with graduate students and postdocs in BCS or other department who have ongoing research projects in research related to subject topics, and develop a class project that synergizes with their research. You can consult with the instructor and TAs to get guidance in conceptualizing, formalizing, and implementing your model; finding suitable datasets; and designing appropriate experiments.

2 Recipe for a good original research paper

The goal of the class project is to carry out a **good original research project** and to write a **good original research paper** describing the project. The recipe for a good original research **project** is straightforward:

- Identify a research question that is important, and whose answer is not yet well understood;
- Develop a plan for work that offers prospects of advancing our understanding of the answer to the question;
- Carry out the work plan to obtain, analyze, and interpret results;
- Take stock of how our (that is, the field's) understanding of the research question should change in light of your results.

The recipe for a good original research **paper** closely corresponds to this:

- State your research question, making clear why it is important (framing in the broader context) and our state of understanding of the answer (related work);
- Explain your approach and how it offers prospects of advancing our understanding of the answer to the question (hypotheses/models/datasets/methods/experiments/planned analyses/predictions);
- Describe the results your approach (results & discussion);
- Revisit the original research question in light of your results (general discussion/conclusion).

Tracking your work against these recipes will help **guide you in doing good science and engineering**, and help

3 Length & style guidelines for the writeup

The length and style of the paper report of your project should follow from the content of your project, but a rough rule of thumb would be a length comparable to a proceedings paper for a conference in computational linguistics (ACL, NAACL, EACL, EMNLP; long-paper format), cognitive science (the Cognitive Science Society conference), machine learning (NeurIPS, ICLR), or linguistics (e.g., NELS, BLS). (Note that these conferences have considerably different page lengths, but due to formatting differences the average quantity of content is not massively different. You can get a sense of the length of the papers in these conferences by browsing papers in the ACL Anthology or recent Cognitive Science Society Conference proceedings, or dropping some text into one of the templates for ACL (look under "Paper Submission and Templates" or CogSci) Some of these conferences have submission deadlines within a couple of months after class projects are due due—including ACL (submission deadline not yet announced, but probably in January or February) CogSci (submission deadline beginning of February). So a successful course project can potentially be submitted to a conference soon afterwards. Shen et al. (2018) and Wilcox et al. (2018) both had origins in 9.190 class projects during 2018.

Make sure you provide enough plain-English context that it is easy for us to understand the problem that you are addressing at a conceptual level, and that the key scientific and/or engineering questions are clear. Figures, and often pseudocode, are generally helpful to include in your writeup. You are also welcome to append your complete implemented code to the paper (this doesn't count toward paper length), but you cannot expect the reader to consult your raw code in understanding your writeup.

4 Organizing your project

We strongly encourage you to work in small groups (2–3 is generally ideal). You might consider using the course's Piazza site to start posting ideas you have for potential class projects and to find fellow seminar participants with whom you share interests for a potential class project. Milestones and recommendations for class projects:

- **September–early October:** brainstorm possible course-project topics and discuss them with fellow class participants, in person and via Piazza. Find others to work with in a group on your project, if you can!
- **Friday, October 20**: your group must turn in a 1-page summary plan. This plan should include summary of the planned project (think of it as something like a 1-page conference abstract). If there will be substantially differentiated roles among group members, also include a brief description of the roles of all group members.
- By **Friday, November 3**, each group should hold a meeting with the instructor or a TA to discuss the 1-page summary and get feedback. (You are welcome—and encouraged!—to turn in your 1-page summary and schedule your meeting considerably earlier than these milestone deadlines.)
- **Friday, December 8:** papers for class projects are due. If you need an extension (e.g., to the end of finals week), please contact us.

5 Example project ideas

- Investigate some aspect of speaker choice (e.g., what principles govern a speaker's word order preferences, or when a speaker omits an optional function word like *that*?), make theoretically motivated predictions regarding likely patterns of behavior, and test these predictions by analyzing a corpus and/or through a behavioral experiment (e.g., Levy & Jaeger, 2007, Zhan & Levy, 2018)
- Construct a model of scrambled-word sentence interpretation or some other comprehension phenomenon that might be modeled using noisy-channel techniques, evaluate its performance, and compare with human performance. You might use weighted finite-state models and/or probabilistic grammars for this purpose, you might use RNNs, or you might use some combination.
- Design an artificial language learning experiment that might test humans' propensity/ability to identify and generalize a distributional pattern in linguistic input, construct a model that makes predictions for behavior in different experimental conditions, and test the model in a behavioral study (e.g., Culbertson et al., 2012)
- Investigate some phenomenon in pragmatic language interpretation through modeling and experimentation (e.g., Bennett & Goodman, 2018)
- Investigate differences in inductive bias among different model classes for linguistic sequence learning (probabilistic grammars, RNNs, n-grams, other) through theoretical analysis and/or by fitting different models to some dataset and comparing their predictions (e.g., Futrell et al., 2018, Gulordava et al., 2018, Linzen et al., 2016).

- Assess how well a human psychological behavior—e.g., word-word free association—can be modeled using distributional semantic models such as `word2vec`, Latent Dirichlet Allocation, or contextualized word embedding models like BERT (Devlin et al., 2018) or GPT-2 (Radford et al., 2019), and compare the differences among models in what aspects of human performance they best capture (e.g., Cattle & Ma, 2017, Shen et al., 2018)
- Design and test a new architecture integrating symbolic grammatical representations and neural generalization (compare Choe & Charniak, 2016 and Dyer et al., 2016)
- Use an NLP model to develop a behavioral psycholinguistic experiment (e.g., Boyce et al., 2020).
- Assess the ability of a psycholinguistic model estimated using NLP tools to quantitatively fit human language processing data (e.g., van Schijndel & Linzen, 2020, Wilcox et al., 2021)
- Develop and/or test a probing technique for decoding linguistic structure from a neural language model (e.g., **tucker-etal:2021-what-if**, which originated as a 9.190 class project)

References

Bennett, E. D., & Goodman, N. D. (2018). Extremely costly intensifiers are stronger than quite costly ones. *Cognition*, *178*, 147–161.

Boyce, V., Futrell, R., & Levy, R. (2020). Maze made easy: Better and easier measurement of incremental processing difficulty. *Journal of Memory and Language*, *111*, 1–13.

Cattle, A., & Ma, X. (2017). Predicting word association strengths, In *Proceedings of the 2017 conference on empirical methods in natural language processing*.

Choe, D. K., & Charniak, E. (2016). Parsing as language modeling, In *Proceedings of the 2016 conference on empirical methods in natural language processing*.

Culbertson, J., Smolensky, P., & Legendre, G. (2012). Learning biases predict a word order universal. *Cognition*, *122*(3), 306–329.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Dyer, C., Kuncoro, A., Ballesteros, M., & Smith, N. A. (2016). Recurrent Neural Network Grammars, In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Futrell, R., Wilcox, E., Morita, T., & Levy, R. (2018). RNNs as psycholinguistic subjects: Syntactic state and grammatical dependency.

Gulordava, K., Bojanowski, P., Grave, E., Linzen, T., & Baroni, M. (2018). Colorless green recurrent networks dream hierarchically, In *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, Louisiana, Association for Computational Linguistics.

Levy, R., & Jaeger, T. F. (2007). Speakers optimize information density through syntactic reduction, In *Proceedings of the 20th conference on Neural Information Processing Systems (NIPS)*.

Linzen, T., Dupoux, E., & Goldberg, Y. (2016). Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners.

Shen, J., Hofer, M., Felbo, B., & Levy, R. (2018). Comparing models of associative meaning: An empirical investigation of reference in simple language games, In *Proceedings of the Twenty-Second Conference on Computational Natural Language Learning (CoNLL)*.

van Schijndel, M., & Linzen, T. (2020). *Neural network surprisal predicts the existence but not the magnitude of human syntactic disambiguation difficulty* [PsyArxiv preprint of May 14, 2020]. PsyArxiv preprint of May 14, 2020.

Wilcox, E., Levy, R. P., Morita, T., & Futrell, R. (2018). What do RNN language models learn about filler–gap dependencies? In *Proceedings of the workshop on analyzing and interpreting neural networks for NLP*.

Wilcox, E., Vani, P., & Levy, R. P. (2021). A targeted assessment of incremental processing in neural language models and humans, In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics*.

Zhan, M., & Levy, R. (2018). Comparing theories of speaker choice using a model of classifier production in Mandarin Chinese, In *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.