CSE 517, Winter 2013: Final Project Instructions

Proposal due: Friday, Feb 22 at 5pm

Final report due: Friday, Mar 22nd at 5pm

Instead of a final exam, you should complete a project; either a literature review or research mini-project. If you choose the research option, we will omit your lowest scored homework assignment from the final grade. Whichever you choose, you will have to submit a proposal and a final report.

1 Option 1: Research Mini-Project

A research mini-project should require approximately as much work as two of our previous homework assignments. But be careful, the amount of work is difficult to determine in advance! Be sure to design a work plan that allows for frequent (ideally weekly) intermediate results that can be discussed in your write up.

Project Ideas

- Implement an algorithm that we didn't have time to cover from a NLP book or research paper (see the non-exhaustive list below). As a intermediate step, be sure to demonstrate interesting learning behavior on toy or simulated data. Here, you might explore issues such as inference, representation and learning. A further goal would be replicate the results from a paper, but this can be surprisingly difficult to achieve in practice.
- Apply an existing algorithm to a new problem. In this case, you would be welcome to use data from your own research. A strong project would carefully describe the new problem, why the application is appropriate, the results achieved, and include a summary of what was learned from the exercise. Negative results can be interesting if you provide a solid argument about why you originally thought the approach would work.
- Conduct an original, but very limited scope, research project. We especially encourage you to be creative about designing projects that can contribute to your research efforts outside of class. Talk to us if you need help coming up with ideas!

Proposal Submit a short paragraph describing your proposed project. Please come to office hours, or contact us if you need help deciding on a topic.

Final Submission A final project report and a single compressed file containing source code with instructions describing how it should be run. The project report should be in PDF format with no more than 6 pages of primary content. You are allowed unlimited space for the citations and appendices, starting on page 7, but your story should be complete and understandable without reading this extra material.

Working in Teams If you choose to do a mini-project, you may work in teams of two. However, if you choose to work in pairs, be aware that we expect twice the work. The page limit for the final report for teams is 8 (instead of 6), excluding citations and appendices. This option is not available if you choose to do a literature review.

2 Option 2: Literature Review

Select an important problem in NLP and conduct a literature review of important papers. Your review should cover 4-5 papers. In your discussion, you should describe each paper and also compare and contrast them. Your report should show insight into the problem and current approaches. This can be done through comparing the chosen papers and discussing their limitations.

Potential Problems Your literature review can discuss one of the problems we saw in class (although you should make sure there's a real added value in your discussion, simply repeating what was said in class is not acceptable). Other potential topics include, but not limited to: named-entity recognition, coreference resolution, information extraction, paraphrase learning and textual entailment. A good approach to finding more potential problems is to explore the papers submitted to top NLP conferences over the last few years. The top conference are ACL, NAACL and EMNLP. All the papers are available in the ACL Anthology (http://aclweb.org/anthology-new/).

Proposal Submit a short paragraph describing the problem you will survey. Be sure to list the papers you will discuss (including URLs to their PDFs).

Final Submission Submit a literature review discussing the papers you selected. You should describe the problem you are discussing and each paper. A good survey will include a deeper discussion of the limitations of each paper and a comparison of the approaches. A better survey will also include well reasoned ideas addressing these limitation. After reading your report we should be convinced that you have deep understanding of the problem and current approaches to it. The project report should be in PDF format with no more than 6 pages of primary content. You are allowed unlimited space for the citations and appendices, starting on page 7, but your story should be complete and understandable without reading this extra material.

Example Paper Lists

- Coreference resolution: Ng (2010), Raghunathan et al. (2010), Ratinov and Roth (2012), Ng (2008) and Wick et al. (2012).
- Information extraction: Carlson et al. (2010), Fader et al. (2011), Hoffmann et al. (2010) and Wu and Weld (2010).
- Textual entailment: Lin and Pantel (2001), Szpektor and Dagan (2008), Berant et al. (2011) and Schoenmackers et al. (2010).
- Any other list of your choice! Talk to us for suggestions or confirmation if you aren't sure.

References

Berant, J., Dagan, I., and Goldberger, J. (2011). Global learning of typed entailment rules. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 610–619. Association for Computational Linguistics.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E., and Mitchell, T. (2010). Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*, volume 2, pages 3–3.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Hoffmann, R., Zhang, C., and Weld, D. (2010). Learning 5000 relational extractors. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL*, volume 10, pages 286–295.

Lin, D. and Pantel, P. (2001). Dirt@ sbt@ discovery of inference rules from text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 323–328. ACM.

Ng, V. (2008). Unsupervised models for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 640–649. Association for Computational Linguistics.

Ng, V. (2010). Supervised noun phrase coreference research: The first fifteen years. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1396–1411. Association for Computational Linguistics.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. (2010). A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501. Association for Computational Linguistics.

Ratinov, L. and Roth, D. (2012). Learning-based multi-sieve co-reference resolution with knowledge. *EMNLP*.

Schoenmackers, S., Etzioni, O., Weld, D., and Davis, J. (2010). Learning first-order horn clauses from web text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1088–1098. Association for Computational Linguistics.

Szpektor, I. and Dagan, I. (2008). Learning entailment rules for unary templates. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 849–856. Association for Computational Linguistics.

Wick, M., Singh, S., and McCallum, A. (2012). A discriminative hierarchical model for fast coreference at large scale. In *Association for Computational Linguistics (ACL)*.

Wu, F. and Weld, D. (2010). Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127. Association for Computational Linguistics.