CIS 472/572, Winter 2013
Final Project
PROPOSAL DUE: Friday, February 22nd at 11:00pm
PAPER DUE: Tuesday, March 19th at 11:00pm

1 Description

Instead of a final exam, CIS 472/572 has a final project which counts for 30% of the grade. It is intended to provide realistic experience in using or researching machine learning.

There are many ways to do the project. The most straightforward is to identify an interesting problem, collect data, design a feature representation, apply several machine learning algorithms (being careful not to train on test data), and analyze the results. Another option is to develop and evaluate a new algorithm. This is more difficult, but can lead to advances in machine learning.

If you want help picking a project, feel free to ask me questions. It’s best if you already have some idea of what you want to do.

2 Methods and Results

Your project must contain theoretical or empirical results. Coming up with new theoretical results of interest is difficult, so I expect that most of you will only present empirical results.

For an application paper, you should evaluate and justify the choices you made. Here are some questions to think about:

- How did you select your data? How much data? What cleaning or processing did you do to the data. (For some problems, you may need to be creative about integrating data from multiple sources, or making do with noise labels.)
- What features did you select and why?
- What algorithms did you use? (You should almost always use more than one, in order to have a comparison.)
- What baselines did you use (if proposing a novel algorithm or feature set or problem formulation)?
- How did you set up the training/tuning/testing data? Did you do cross-validation? How did you tune the parameters?

• Which algorithm performs best? Can you determine why that algorithm works best?

You do not need to implement everything yourself. Weka and scikit-learn are popular open-source toolkits that already include many common classifiers. Please do follow the scientific method. Develop appropriate experiments to validate or refute your hypotheses, as well as to provide more insight. This does not need to be publishable research, but it should demonstrate that you understand how to apply machine learning to a real problem (for an application paper) or how to develop and evaluate novel algorithms (for an algorithms paper).

Negative results are acceptable. If you get a negative result, explore what happened and why. Not enough data? Overfitting? Bad features? Noisy labels? Different distribution at test time? Explore what led to the poor results and try to determine if that could be overcome.

3 Writing

All papers will be expected to be clearly written with a good structure, but graduate students and larger groups will be held to a higher standard than undergraduates and smaller groups. Many machine learning papers use a structure similar to the following:

1. Abstract: Summarize the entire paper (including results) in 50-250 words.

2. Introduction: Identify the problem you’re trying to solve, describe why it’s important, and outline the key method or strategy that you will use to solve it.

3. Background: Describe the technologies or ideas that you will build on in your method. For an application paper, this could simply be a detailed description of the problem you’re trying to solve. For an algorithm paper, this could be the machine learning methods that you’re extending.

4. Methods: Describe your approach to solving the problem. This should contain your key contributions.

5. Experiments: Evaluate your approach experimentally. Describe your methods in enough detail that another researcher could replicate them. How well does your method work? Does your method outperform reasonable baselines? How does your method compare to simplified versions of your method? What kinds of errors remain? What interesting things do you learn from your experiments? Tables of results are useful, but charts and figures are often better.
6. Conclusion: Summarize your contributions and discuss future work (50-500 words).

7. References: Works that you cite in the body of your paper. You may use any standard citation style as long as it is consistent.

I recommend that you use a structure similar to this one, unless you have a good reason.

I do not require perfect English, but I greatly appreciate clear writing. Your conclusions should be supported by evidence. Your arguments should follow logically. Each paragraph should discuss a single idea. If you’re having trouble, there is writing tutoring available on campus for all students.

Learning to write a good technical paper is an extremely valuable skill in both graduate school and industry. Writing well is very difficult, even for experienced writers, but it does get easier with practice.

4 Proposal

In order to give you early feedback on your ideas, please send me a 1-page proposal by Friday, February 22nd. You only need to send one proposal per group. Your proposal should describe the problem you are trying to solve and your ideas for how to solve it. What data will you use? What features will you use? What algorithms will you try? What metrics will you apply? You do not need to know everything, but you should have a reasonable plan. You are free to deviate from that plan based on the results you obtain or any new ideas that you come up with.

5 What To Turn In

Please submit everything to lowd@cs.uoregon.edu.