USC Viterbi

CSCI 699: Topics in Discrete Optimization and Learning

Units: 4
Fall 2018 – Monday – 2:00-5:20

Mondays 2:00- 5:20pm
Location: KAP 141

Instructors: Bistra Dilkina
Office: SAL 304
Office Hours: TBD

Contact Info: dilkina@usc.edu

Teaching Assistant: TBD

Course Description

This course will examine recent research trends in leveraging machine learning in computational methods for solving NP-hard discrete optimization problems. The first part of the course will introduce students to computational discrete optimization and in particular local search and branch and bound. The main part of the course will study in depth recent papers focusing on a data-driven algorithm design for combinatorial problems, where ML techniques are used to improve existing methods or design new ones altogether. This class will be targeted at PhD students. Background in ML is expected. Mathematical maturity, as well as research experience in computer science and/or data science is strongly recommended.

Learning Objectives

1. Learn solution techniques for discrete optimization
2. Learn how to empirically evaluate computational performance of solvers.
3. Learn how to use different ML techniques in the context of discrete optimization
4. Learn how to critically read a technical research paper on this topic.

.

Prerequisite(s): sufficient mathematical background; some background in AI, machine learning and discrete optimization, with research experience in at least one, is strongly recommended; good programming skills

Recommended Preparation: CS Algorithms class (preferably at a graduate level) is strongly advisable, ML and optimization course work or background that is advisable, not mandatory

Course Notes

Lecture slides/notes will be available online after class.

Required Readings and Supplementary Materials

The course will not have any official textbook but will instead use assigned papers and book chapters reading. This is a preliminary list of the reading list:

Computational Testing

Constraint Integer Programming, Tobias Achterberg, Ph. D. Thesis, Berlin 2009. Beautiful examples of careful computational testing.

Experimental analysis of algorithms, Catherine McGeoch. Notices of the AMS, March 2001.

CH. 5, STOCHASTIC LOCAL SEARCH: FOUNDATIONS AND APPLICATIONS by Holger H. Hoos and Thomas Stützle

SAT:
Learning a SAT Solver from Single-Bit Supervision. Daniel Selsam, Matthew Lamm, Benedikt Bunz, Percy Liang, Leonardo de Moura, David L. Dill, ArXiv 2018

CSP:
Learning Robust Search Strategies Using a Bandit-Based Approach. Wei Xia, Roland H. C. Yap, AAAI 2018

ML for Branch-and-Bound (Week 7 + 8)
On learning and branching: a survey, A. Lodi, G. Zarpellon.
MIP Branching:
Learning to branch in mixed integer programming, E. B. Khalil, P. Le Bodic, L. Song, G. Nemhauser, B. Dilkina, AAAI 2016
Learning to Branch, Maria-Florina Balcan, Travis Dick, Tuomas Sandholm, Ellen Vitercik, ICML 2018
DASH: Dynamic Approach for Switching Heuristics. Liberto, G. Di, S. Kadioglu, K. Leo, Y. Malitsky. European Journal of Operational Research 248(3) 943–953, 2016.
MIP Node Selection:
Learning to Search in Branch-and-Bound Algorithms, He He, Hal Daume III, Jason Eisner. Advances in neural information processing systems, 3293-3301, 2014.
Learning to Search via Retrospective Imitation, Jialin Song, Ravi Lanka, Albert Zhao, Yisong Yue, Masahiro Ono. ArXiv, 2018
Guiding Combinatorial Optimization with UCT, Sabharwal, Ashish, Horst Samulowitz, and Chandra Reddy. CPAIOR, 2012.
Other MIP components:
Learning to run heuristics in tree search. Khalil, E. B., B. Dilkina, G. L. Nemhauser, S. Ahmed, Y. Shao. IJCAI 2017.
Learning when to use a decomposition. Kruber, M., M. E. Lubbecke, A. Parmentier. CPAIOR 2017.

Q-Learning
Learning combinatorial optimization algorithms over graphs, H. Dai, E. B. Khalil, Y. Zhang, B. Dilkina, L. Song. NIPS 2017.

Sequence-to-Sequence Learning
Pointer networks, O. Vinyals, M. Fortunato, N. Jaitly. 2017.
Neural combinatorial optimization with reinforcement learning,I. Bello, H. Pham, Q. V. Le, M. Norouzi, S. Bengio. 2017. Lecture by Samy Bengio (Berlin, June 2017.)
Deep reinforcement learning for solving the vehicle routing problem, M. Nazari, A. Oroojlooy, L. V. Snyder, M. Takac. February 2018.
Background:
Neural machine translation by jointly learning to align and translate, D. Bahdanau, K. Cho, Y. Bengio. 2014.
Sequence to sequence learning with neural networks, I. Sutskever, O. Vinyals, Q. V. Le. 2014.

Attention
Attention is all you need, A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, I. Polosukhin. December 2017.
Attention solves your TSP, W. Kool, M. Welling. March 2018. (Source code on GitHub.)

ML for Algorithm Selection
Algorithm Selection using Reinforcement Learning. Lagoudakis, Michail G., and Michael L. Littman. ICML. 2000.
Deep Learning for Algorithm Portfolios. Andrea Loreggia, Yuri Malitsky, Horst Samulowitz, Vijay Saraswat. In AAAI, 2016

Feature-Based Algorithm Selection for Mixed Integer Programming. Alexander Georges and Ambros Gleixner and Gorana Gojic and Robert Lion Gottwald and David Haley and Gregor Hendel and Bartlomiej Matejczy. Technical Report, 2018
Algorithm selection for combinatorial search problems: A survey. L Kotthoff. Data Mining and Constraint Programming, 149-190, 2016
ParamILS: An Automatic Algorithm Configuration Framework, JAIR 2009
Sequential model-based optimization for general algorithm configuration. Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. In Proceedings of the 5th international conference on Learning and Intelligent Optimization (LION'05), Carlos Coello Coello (Ed.). Springer-Verlag, Berlin, Heidelberg, 507-523, 2011

Use of MIP in ML

Deep neural networks and mixed integer linear optimization. Fischetti, M. & Jo, J. Constraints (2018). https://doi.org/10.1007/s10601-018-9285-6

Description and Assessment of Assignments

1. Reviews: We will explore the course topics through a series of assigned readings in the form of papers (and book chapters). Students will be expected to read the papers before class and submit a one page review for 2 of the assigned reading papers as homework. Students will get credit for every review submitted. Reviews are expected to be turned in at the beginning of the. Every review should address the following 5 questions:
   1. What is the main problem/task addressed by the paper?
   2. What was done before, and how does this paper improve on it?
   3. What is the one cool technique/idea/finding that was learned from this paper?
   4. What part of the paper was difficult to understand?
   5. What generalization or extension of the paper could be done?

2. Paper Presentation: Students will present a recent research paper relevant to the course topic and lead discussions with the class.
3. Project: A key component of the course will be to get a hands on experience in using ML in the context of discrete optimization. Projects will be alone or in small teams. Projects will be graded based on their novelty and technical results. Students will be expected to prepare project proposals halfway through the course, and give presentations of their projects/papers at the end of the course. The final project paper should have the structure of a conference paper with problem statement, lit review, approach, empirical results and discussion. A statement of author contributions (i.e. who did what) must be turned in with the final draft. Rough drafts and partial drafts will be due at different points throughout the semester so that the instructor may provide students with constructive input along the way.

Grading Breakdown

	Weight
Class participation	**5%**
Paper Reviews (10 papers @ 1% each)	**10%**
Paper Presentation	**25%**
Final Project	**60%**
Project Proposal (5%)	
Preliminary Paper (10%)	
Final Presentation (10%)	
Project/Final Paper (35%)	

Assignment Submission Policy

Paper reviews and other deliverables should be submitted before the start of class on the due date (unless otherwise instructed).

Course Schedule: A Weekly Breakdown

	Topics/Daily Activities	Readings and Homework	Deliverable/ Due Dates
Week 1 Dates	Intro to Discrete Optimization: NP-hard problems; TSP, VC and other NP-hard graph optimization problems; SAT; CSP; MIP; Branch and Bound; Local Search		
Week 2 Dates			
Week 3 Dates			
Week 4 Dates		**Paper Reviews**	
Week 5 Dates	RL for Opt	**Paper Reviews**	
Week 6 Dates	RL for Opt	**Paper Reviews**	**Project Group Formation**
Week 7 Dates	ML for MIP Branch and Bound	**Paper Reviews**	
Week 8 Dates	ML for MIP Branch and Bound	**Paper Reviews**	**Project Proposal**
Week 9 Dates	ML for MIP Branch and Bound	**Paper Reviews**	
Week 10 Dates	DL for Discrete Opt	**Paper Reviews**	
Week 11 Dates	DL for Discrete Opt	**Paper Reviews**	
Week 12 Dates	ML for Algorithm Selection	**Paper Reviews**	
Week 13 Dates	ML for Algorithm Selection	**Paper Reviews**	**Project prelim paper**
Week 14 Dates	Use of MIP in ML	**Paper Reviews**	
Week 15 Dates	Project Presentation		**Project Presentation**
FINAL Date			**Final project paper** Date: For the date and time of the final for this class, consult the USC *Schedule of Classes* at www.usc.edu/soc.

Statement on Academic Conduct and Support Systems

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Section 11, *Behavior Violating University Standards* https://scampus.usc.edu/1100-behavior-violating-university-standards-and-appropriate-sanctions. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Equity and Diversity* http://equity.usc.edu or to the *Department of*

Public Safety http://capsnet.usc.edu/department/department-public-safety/online-forms/contact-us. This is important for the safety of the whole USC community. Another member of the university community – such as a friend, classmate, advisor, or faculty member – can help initiate the report, or can initiate the report on behalf of another person. *The Center for Women and Men* http://www.usc.edu/student-affairs/cwm/ provides 24/7 confidential support, and the sexual assault resource center webpage http://sarc.usc.edu describes reporting options and other resources.

Note on Collaborative Work

For collaborative projects, students are expected to have equal distribution. If there is any perceived imbalance in the collaborative project, the student should bring this to the attention of the instructor or the teaching assistant.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* *http://emergency.usc.edu* will provide safety and other updates, including ways in which instruction will be continued by means of blackboard, teleconferencing, and other technology.