Human language is an intricate system; each sentence has its own grammatical structure, interconnected references, and set of possible meanings. The field of Natural Language Processing (NLP) aims to build computational models of language in order to make predictions based on real-world textual data. Example applications of NLP include machine translation, information extraction, and question answering. Tools developed for these problems are increasingly becoming part of daily life, from speech and dialogue systems on mobile devices to structured search on the web to real-time translation. NLP is a rich intersection of formal modeling, applied algorithms and scalable data systems, and has served as an important application domain for related fields such as Machine Learning (ML).

The central focus of my dissertation is the application of models and algorithms developed for combinatorial optimization to the construction of large-scale systems for processing textual data. I have focused on three areas: combinatorial optimization for natural language inference, NLP applications, and efficient inference at scale. There has been a natural feedback loop among each of these three areas. Domain knowledge of the application under investigation helps me understand the modeling choices necessary when developing an algorithm, which in turn are influenced by the practical efficiency concerns observed on empirical data.

Combinatorial Optimization for Natural Language Inference

Statistical NLP tasks are often framed as supervised learning problems, i.e. a statistical model is trained from a corpus of manually annotated text. However unlike binary classifiers, the models predict a *structured* output based on the input; for instance, the best sequence of part-of-speech tags for an English sentence or the best translation for a foreign-language sentence. Since the set of possible output structures is too large to be enumerated, this prediction requires solving a combinatorial optimization problem. For large-scale NLP systems the bottleneck for training and decoding is the ability to solve this inference problem efficiently. Unfortunately, for many complex models the corresponding inference problem is difficult or even NP-Hard to solve, and in practice many systems rely heavily on heuristic search. My work argues that while these problems are hard in the worst-case, in examples from real language we can often find *optimal* solutions efficiently.

The main tool of this work is Lagrangian relaxation, a classical technique from combinatorial optimization. Lagrangian relaxation specifies a simple iterative algorithm that may produce a certificate of optimality, even if the underlying problem is hard in the worst case. Qualitatively, the method utilizes simple relaxations of the problem that are easy to solve, but capture important structure of the underlying problem. Formally, it consists of subgradient optimization of the dual of an integer linear programming representation of the original inference problem. The guarantees of the method can be derived from this interpretation.

My work was the first to apply Lagrangian relaxation to natural language inference, and to show that for important NLP problems it often finds exact solutions with a certificate of optimality [10]. This work motivated many other natural language researchers to apply Lagrangian relaxation, and the closely related dual decomposition algorithm, to core NLP tasks, including: translation alignment [2]; event extraction [5]; semantic parsing [1]; inference with weighted automata [4]; and many other applications. I have also given tutorials on the method at a major NLP conference, [11], at a major ML conference, [12], and written a journal tutorial on the technique [13].

NLP Applications

After introducing Lagrangian relaxation as an inference method for NLP, I applied it and related methods to several important application areas. The challenge of applying these combinatorial optimization methods within NLP is developing relaxations for difficult inference problems that capture important properties of the underlying problem, yet are still efficient to run on full-scale data. These applications include:

- **Parsing.** Non-projective dependency grammars are an important formalism for many free-word order languages such as Czech and Dutch. For simple models inferring the best dependency structure can be done in polynomial-time using the maximum weight directed spanning tree algorithm. Unfortunately, for richer statistical models inferring the best dependency structure is NP-Hard. My work shows that a simple finite-state relaxation can capture much of the important model structure, and that, when used with Lagrangian relaxation it can find optimal solutions with certificates for real-world examples. The implementation of this method produces optimal solutions and state-of-the-art results for non-projective languages [3]. This work won the best paper award at EMNLP 2010.

- **Machine Translation.** Statistical machine translation is the problem of predicting the best translation for an input sentence based on a statistical model. Making this prediction is computationally difficult; for example for the most commonly used model inference is hard under a reduction to the traveling-salesman problem. For this reason most systems, including large-scale systems like Google Translate, rely on heuristic beam search for inference. My work on translation inference shows that for a syntax-based model using a simple dynamic programming algorithm combined with classical all-pairs shortest path yields an efficient and almost always optimal relaxation algorithm for translation [7]. In follow-up work, I show that this relaxation, as well as a different phrase-based relaxation, can be used to extend the widely applied beam-search heuristic in order to find exact solutions even more efficiently [6]. These papers show that a problem widely assumed to require a heuristic algorithm can be solved efficiently and optimally with a simple combinatorial algorithm.

- **Joint and Document-Level Models.** Due to the complexity of inference, many NLP systems make standard independence assumptions about language, including: (1) assuming simpler structures are independent of more complex structures, e.g. part-of-speech tags are predicted before parsing, (2) assuming sentences in a document are independent of each other. Often times it is preferable to loosen these assumptions. My work looks at two classical tasks in NLP – combining multiple parsers and combining tagging and parsing – and develops a dual decomposition algorithm that exactly solves the joint problem by repeatedly solving versions of the independent problems, and shows an improvement in parsing accuracy [10]. A similar method can be used to modeling the relationship between sentences. My work constructs a model with document-level constraints that can be solved with only minor changes to existing single-sentence algorithms [9].

Efficient Large-Scale Decoding Systems

Another area of interest, particularly for practical, large-scale systems, is constructing highly-efficient inference algorithms that allow for a small drop in prediction accuracy.

While attending graduate school, I also worked part-time at Google Research in New York. At Google, I initiated a project to build a highly-optimized framework for efficient structured prediction. I built an end-to-end system for training structured models and performing inference. The framework allows researchers to experiment with different exact and approximate optimization methods, as well as a variety of learning algorithms.

I used this framework to reproduce a state-of-the-art higher-order dependency parser. In order to optimize this parser for speed I implemented two approximate methods: (1) a simplified parser known as a Vine Parser to act as a linear-time, first-pass pruning model; (2) a training method known as Structured Prediction Cascades designed specially for training pruning models. The final model maintains nearly the same parsing accuracy as the state-of-art system but at 200-times the speed [8]. The system made an impractical parsing technique usable on large-scale data. The work also won the best paper award at NAACL 2012. After my internship the framework continues to be used at Google and is the basis for several additional papers.

Future Work

Natural Language Processing is at an intriguing point as a field. Development of new algorithms along with a massive increase in the amount of available textual data has enabled researchers to create accurate real-world systems for tasks that were originally small-scale in nature. A major challenge for NLP researchers is to develop models and algorithms that can continue to utilize the increasing amount of data in order to further improve the accuracy of language systems.

I am excited to work in NLP to take up this challenge. One problem that interests me is developing *unified* tools for high-level NLP tasks, such as language understanding. Work in NLP traditionally focuses on specific domain areas, such as parsing or coreference; however as we begin to work on higher-level problems and utilize much larger textual resources, it will be important to build models that make joint predictions based on multiple aspects of language. A joint system might predict syntax, semantics, and discourse structure simultaneously and consider the interplay between the different aspects. This joint prediction problem is challenging from a modeling, inference, and systems-building perspective.

Another area I am interested in is unsupervised learning for NLP. Much of my past research has focused on supervised problems using medium-size, manually annotated data sets. With the increase of available data, it is increasingly important to make use of the large amount of text without supervised annotations. NLP researchers have traditionally relied heavily on heuristic methods like EM for unsupervised learning. I am interested in developing and utilizing alternative methods for learning from unsupervised textual data, with the goal of providing efficient, accurate algorithms that can also give improved theoretical guarantees.

Finally, from a practical perspective, I am excited to develop open-source software for working with large-scale textual data. In addition to my experience at Google, I have also worked for several years as a software engineer at Facebook. These experiences have helped me understand how to develop efficient, large-scale software with a team. One of my goals is to apply this background to develop open research systems that can be used to solve real-world language processing problems.

References

[1] Das, D., Martins, A., and Smith, N. An exact dual decomposition algorithm for shallow

semantic parsing with constraints. *Proceedings of* SEM.[ii, 10, 50]* (2012).

[2] DeNero, J., and Macherey, K. Model-Based Aligner Combination Using Dual Decomposition. In *Proc. ACL* (2011).

[3] Koo, T., Rush, A. M., Collins, M., Jaakkola, T., and Sontag, D. Dual decomposition for parsing with non-projective head automata. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (Cambridge, MA, October 2010), Association for Computational Linguistics, pp. 1288–1298.

[4] Paul, M. J., and Eisner, J. Implicitly intersecting weighted automata using dual decomposition. In *Proc. NAACL* (2012).

[5] Riedel, S., and McCallum, A. Fast and robust joint models for biomedical event extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (Edinburgh, Scotland, UK., July 2011), Association for Computational Linguistics, pp. 1–12.

[6] Rush, A., Chang, Y., and Collins, M. Optimal Beam Search for Machine Translation. In *Proc. EMNLP* (2013).

[7] Rush, A., and Collins, M. Exact Decoding of Syntactic Translation Models through Lagrangian Relaxation. In *Proc. ACL* (2011).

[8] Rush, A., and Petrov, S. Vine pruning for efficient multi-pass dependency parsing. In *Joint Human Language Technology Conference/Annual Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL12)* (2012).

[9] Rush, A., Reichart, R., Collins, M., and Globerson, A. Improved parsing and pos tagging using inter-sentence consistency constraints. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (Jeju Island, Korea, July 2012), Association for Computational Linguistics, pp. 1434–1444.

[10] Rush, A., Sontag, D., Collins, M., and Jaakkola, T. On Dual Decomposition and Linear Programming Relaxations for Natural Language Processing. In *Proc. EMNLP* (2010).

[11] Rush, A. M., and Collins, M. Dual decomposition for natural language processing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011* (2011), Association for Computational Linguistics, p. 6.

[12] Rush, A. M., and Collins, M. Lagrangian relaxation for natural language processing. In *Proc. NIPS. Tutorial Abstracts* (2011), NIPS.

[13] Rush, A. M., and Collins, M. A tutorial on dual decomposition and lagrangian relaxation for inference in natural language processing. *J. Artif. Intell. Res. (JAIR) 45* (2012), 305–362.