ACL HLT 2011

The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies

Proceedings of Tutorial Abstracts

19 June, 2011
Portland, Oregon, USA

Production and Manufacturing by
Omnipress, Inc.
2600 Anderson Street
Madison, WI 53704 USA

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN-139781937284077

Tutorials Chairs:

Andy Way, Dublin City University, Ireland
Patrick Pantel, Microsoft Research, USA

Table of Contents

Conference Program

Sunday, June 19, 2011

Morning Tutorials

9:00–12:30 *Beyond Structured Prediction: Inverse Reinforcement Learning*
Hal Daumé III

9:00–12:30 *Formal and Empirical Grammatical Inference*
Jeffrey Heinz, Colin de la Higuera and Menno van Zaanen

9:00–12:30 *Automatic Summarization*
Ani Nenkova, Sameer Maskey and Yang Liu

Afternoon Tutorials

2:00–5:30 *Web Search Queries as a Corpus*
Marius Paşca

2:00–5:30 *Rich Prior Knowledge in Learning for Natural Language Processing*
Gregory Druck, Kuzman Ganchev and João Graça

2:00–5:30 *Dual Decomposition for Natural Language Processing*
Michael Collins and Alexander M. Rush

Beyond Structured Prediction: Inverse Reinforcement Learning

Hal Daumé III
Computer Science and UMIACS
University of Maryland, College Park
me@hal3.name

1 Introduction

Machine learning is all about making predictions; language is full of complex rich structure. Structured prediction marries these two. However, structured prediction isn't always enough: sometimes the world throws even more complex data at us, and we need reinforcement learning techniques. This tutorial is all about the *how* and the *why* of structured prediction and inverse reinforcement learning (aka inverse optimal control): participants should walk away comfortable that they could implement many structured prediction and IRL algorithms, and have a sense of which ones might work for which problems. I gave a similar tutorial at ACL 2010. The first half of these two tutorials is 90% the same; the second half is only 50% the same.

2 Content Overview

The first half of the tutorial will cover the "basics" of structured prediction: the structured perceptron and Magerman's incremental parsing algorithm. It will then build up to more advanced algorithms that are shockingly reminiscent of these simple approaches: maximum margin techniques and search-based structured prediction.

The second half of the tutorial will ask the question: what happens when our standard assumptions about our data are violated? This is what leads us into the world of reinforcement learning (the basics of which we'll cover) and then to inverse reinforcement learning and inverse optimal control.

Throughout the tutorial, we will see examples ranging from simple (part of speech tagging, named entity recognition, etc.) through complex (parsing, machine translation).

The tutorial does not assume attendees know anything about structured prediction or reinforcement learning (though it will hopefully be interesting even to those who know some!), but *does* assume some knowledge of simple machine learning.

3 Tutorial Outline

Part I: Structured prediction

- What is structured prediction?
- Refresher on binary classification
  - What does it mean to learn?
  - Linear models for classification
  - Batch versus stochastic optimization
- From perceptron to structured perceptron
  - Linear models for structured prediction
  - The "argmax" problem
  - From perceptron to margins
- Search-based structured prediction
  - Stacking
  - Training classifiers to make parsing decisions
  - Searn and generalizations

Part II: Inverse reinforcement learning

- Refersher on reinforcement learning
  - Markov decision processes
  - Q learning
- Inverse optimal control and A* search
  - Maximum margin planning
  - Learning to search
  - Dagger
- Apprenticeship learning
- Open problems

References

See http://www.cs.utah.edu/~suresh/mediawiki/index.php/MLRG/spring10.

Tutorial Abstracts of ACL/HLT 2011, page 1,
Portland, Oregon, USA, June 2011.

Formal and Empirical Grammatical Inference

Jeffrey Heinz
University of Delaware
42 E. Delaware Ave
Newark DE 19716
USA
heinz@udel.edu

Colin de la Higuera
Université de Nantes
2 rue de la Houssinière
44322 Nantes Cedex 03
France
cdlh@univ-nantes.fr

Menno van Zaanen
Tilburg University
P.O. Box 90153
NL-5000 LE Tilburg
The Netherlands
mvzaanen@uvt.nl

1 Introduction

Computational linguistics (CL) often faces learning problems: what algorithm takes as input some finite dataset, such as (annotated) corpora, and outputs a system that behaves "correctly" on some task? Data chunking, dependency parsing, named entity recognition, part-of-speech tagging, semantic role labelling, and the more general problem of deciding which out of all possible (structured) strings are grammatical are all learning problems in this sense.

The subfield of theoretical computer science known as grammatical inference (GI) studies how grammars can be obtained from data. Although there has been limited interaction between the fields of GI and CL to date, research by the GI community directly impacts the study of human language and in particular CL. The purpose of this tutorial is to introduce GI to computational linguists.

2 Content Overview

This tutorial shows how the theories, algorithms, and models studied by the GI community can benefit CL. Particular attention is paid to both foundational issues (e.g. how learning ought to be defined), which are relevant to all learning problems even those outside CL, and issues specific to problems within CL. One theme that runs throughout the tutorial is how properties of natural languages can be used to reduce the instance space of the learning problem, which can lead to provably correct learning solutions. E.g. some natural language patterns are mildly context-sensitive (MCS), but many linguists agree that not all MCS languages are possible natural languages. So instead of trying to learn all MCS languages (or context-free, regular or stochastic variants thereof), one approach is to try to define a smaller problem space that better approximates natural languages. Interestingly, many of the regions that the GI community has carved out for natural language appear to have the right properties for feasible learning even under the most stringent definitions of learning.

3 Tutorial Outline

1. *Formal GI and learning theory.* Those learnability properties which should be regarded as necessary (not sufficient) conditions for "good" language learning are discussed. Additionally, the most important proofs that use similar algorithmic ideas will be described.
2. *Empirical approaches to regular and sub-regular natural language classes.* This focuses on sub-regular classes that are learnable under many (including the hardest) settings. General learning strategies are introduced as well as probabilistic variants, which are illustrated with natural language examples, primarily in the domain of phonology.
3. *Empirical approaches to non-regular natural language classes.* This treats learnability of context-free/sensitive formalisms where the aim is to approximate the grammar of a specific language from data, not to show learnability of classes of languages. An overview of well-known empirical grammatical inference systems will be given in the context of natural language syntax and semantics.

Tutorial Abstracts of ACL/HLT 2011, page 2,
Portland, Oregon, USA, June 2011.

Automatic Summarization

Ani Nenkova
Univ. of Pennsylvania
Philadelphia, PA
nenkova@seas.upenn.edu

Sameer Maskey
IBM Research
Yorktown Heights, NY
smaskey@us.ibm.com

Yang Liu
Univ. of Texas at Dallas
Richardson, TX
yangl@hlt.utdallas.edu

1 Introduction

In the past decade, we have seen that the amount of digital data, such as news, scientific articles, blogs, conversations, increases at an exponential pace. The need to address 'information overload' by developing automatic summarization systems has never been more pressing. At the same time, approaches and algorithms for summarization have matured and increased in complexity, and interest in summarization research has intensified, with numerous publications on the topic each year. A newcomer to the field may find navigating the existing literature to be a daunting task. In this tutorial, we aim to give a systematic overview of traditional and more recent approaches for text and speech summarization.

2 Content Overview

A core problem in summarization research is devising methods to estimate the importance of a unit, be it a word, clause, sentence or utterance, in the input. A few classical methods will be introduced, but the overall emphasis will be on most recent advances. We will cover log-likelihood test for topic word discovery and graph-based models for sentence importance, and will discuss semantically rich approaches based on latent semantic analysis, lexical resources. We will then turn to the most recent Bayesian models of summarization. For supervised machine learning approaches, we will discuss the suite of traditional features used in summarization, as well as issued with data annotation and acquisition.

Ultimately, the summary will be a collection of important units. The summary can be selected in a greedy manner, choosing the most informative sentence, one by one; or the units can be selected jointly, and optimized for informativeness. We discuss both approaches, with emphasis on recent optimization work.

In the part on evaluation we will discuss the standard manual and automatic metrics for evaluation, as well as very recent work on fully automatic evaluation.

We then turn to domain specific summarization, particularly summarization of scientific articles and speech data (telephone conversations, broadcast news, meetings and lectures). In speech, the acoustic signal brings more information that can be exploited as features in summarization, but also poses unique problems which we discuss related to disfluencies, lack of sentence or clause boundaries, and recognition errors.

We will only briefly touch on key but under-researched issues of linguistic quality of summaries, deeper semantic analysis for summarization, and abstractive summarization.

3 Tutorial Outline

1: Computing informativeness
- Frequency-driven: topic words, clustering, graph approaches
- Semantic approaches: lexical chains, latent semantic analysis
- Probabilistic (Bayesian) models
- Supervised approaches

2: Optimizing informativeness and minimizing redundancy
- Maximal marginal relevance
- Integer linear programming
- Redundancy removal

3: Evaluation
- Manual evaluation: Responsivness and Pyramid
- Automatic: Rouge
- Fully automatic

4: Domain specific summarization
- Scientific articles
- Biographical
- Speech summarization
  * Utterance segmentation
  * Acoustic features
  * Dealing with recognition errors
  * Disfluency removal and compression

Tutorial Abstracts of ACL/HLT 2011, page 3,
Portland, Oregon, USA, June 2011.

Web Search Queries as a Corpus

Marius Paşca
Google Inc.
Mountain View, California 94043
mars@google.com

1 Introduction

As noisy and unreliable as they may be, Web search queries indirectly convey knowledge just as they request knowledge. Indeed, queries specify constraints, even if as brittle as the mere presence of an additional keyword or phrase, that loosely describe what knowledge is being requested. In the process of asking "how many calories are burned during skiing", one implies that skiing may be an activity during which the body consumes calories, whereas the more condensed "amg latest album" still suggests that amg may be a musician or band, even to someone unfamiliar with the respective topics. As such, search queries are cursory reflections of knowledge encoded deeply within unstructured and structured content available in documents on the Web and elsewhere.

The notion that inherently-noisy Web search queries may collectively serve as a text corpus is an intriguing alternative to using document corpora. This tutorial gives an overview of the characteristics of, and types of knowledge available in, queries as a corpus. It reviews extraction methods developed recently for extracting such knowledge. Considering the building blocks that would contribute towards the automatic construction of knowledge bases, queries lend themselves as a useful data source in the acquisition of classes of instances (e.g., *palo alto*, *santa barbara*, *twentynine palms*), where the classes are unlabeled or labeled (e.g., *california cities*), possibly organized as hierarchies of search intents; as well as relations, including class attributes (e.g., *population density*, *mayor*).

2 Content Overview

The tutorial covers characteristics of search queries, when considered as an input data source in open-domain information extraction, and their impact on extraction methods operating over queries as opposed to documents; types of knowledge for which queries lend themselves as a useful data source in information extraction; detailed methods for extracting classes, instances and relations from queries; and implications in semantic annotation of queries, understanding query intent, and information access and retrieval in general.

3 Tutorial Outline

. Introduction
. - Overview of knowledge acquisition from text
. - Goals of open-domain information extraction
. - Extraction from documents vs. queries
. Queries as a corpus
. - Intrinsic aspects: distribution, lexical structure
. - Extrinsic aspects: temporality, demographics
. - Beyond individual queries: sessions, clicks
. Methods for knowledge acquisition from queries
. - Extraction of instances and classes
. - Extraction of attributes and relations
. Discussion
. - Implications and limitations
. - Applications

Tutorial Abstracts of ACL/HLT 2011, page 4,
Portland, Oregon, USA, June 2011.

Rich Prior Knowledge in Learning for Natural Language Processing

Gregory Druck
U. of Massachusetts Amherst
gdruck@cs.umass.edu

Kuzman Ganchev
Google Inc.
kuzman@google.com

João Graça
University of Pennsylvania
joao.graca@l2f.inesc-id.pt

1 Introduction

We possess a wealth of prior knowledge about most prediction problems, and particularly so for many of the fundamental tasks in natural language processing. Unfortunately, it is often difficult to make use of this type of information during learning, as it typically does not come in the form of labeled examples, may be difficult to encode as a prior on parameters in a Bayesian setting, and may be impossible to incorporate into a tractable model. Instead, we usually have prior knowledge about the values of output variables. For example, linguistic knowledge or an out-of-domain parser may provide the locations of likely syntactic dependencies for grammar induction. Motivated by the prospect of being able to naturally leverage such knowledge, four different groups have recently developed similar, general frameworks for expressing and learning with side information about output variables.

2 Content Overview

This tutorial describes how to encode side information about output variables, and how to leverage this encoding and an unannotated corpus during learning. We survey the different frameworks, explaining how they are connected and the trade-offs between them. We also survey several applications that have been explored in the literature, including applications to grammar and part-of-speech induction, word alignment, information extraction, text classification, and multi-view learning. Prior knowledge used in these applications ranges from structural information that cannot be efficiently encoded in the model, to knowledge about the approximate expectations of some features, to knowledge of some incomplete and noisy labellings. These applications also address several different problem settings, including unsupervised, lightly supervised, and semi-supervised learning, and utilize both generative and discriminative models.

Additionally, we discuss issues that come up in implementation, and describe a toolkit that provides out-of-the-box support for the applications described in the tutorial, and is extensible to other applications and new types of prior knowledge.

3 Tutorial Outline

1. **Introduction**: prior knowledge in NLP, previous approaches for leveraging, motivation for constraining output variables, demonstration
2. **Learning with Prior Knowledge**: Constraint-Driven Learning (UIUC), Posterior Regularization (UPenn), Generalized Expectation Criteria (UMass Amherst), Learning from Measurements (UC Berkley)
3. **Applications**: document classification (labeled features), information extraction (long-range dependencies), word alignment (symmetry), POS tagging (posterior sparsity), dependency parsing (linguistic knowledge, noisy labels)
4. **Implementation**: implementation guidance, tutorial on existing software packages

4 References

see: http://sideinfo.wikkii.com

Tutorial Abstracts of ACL/HLT 2011, page 5,
Portland, Oregon, USA, June 2011.

Dual Decomposition for Natural Language Processing

Michael Collins
Department of Computer Science
Columbia University
New York, NY, USA
mcollins@cs.columbia.edu

Alexander M. Rush
CSAIL
Massachusetts Institute of Technology
Cambridge, MA, USA
srush@csail.mit.edu

1 Introduction

For many tasks in natural language processing, finding the best solution requires a search over a large set of possible choices. Solving this decoding problem exactly can be complex and inefficient, and so researchers often use approximate techniques at the cost of model accuracy. In this tutorial, we present *dual decomposition* as an alternative method for decoding in natural language problems. Dual decomposition produces exact algorithms that rely only on basic combinatorial algorithms, like shortest path or minimum spanning tree, as building blocks. Since these subcomponents are straightforward to implement and are well-understood, the resulting decoding algorithms are often simpler and significantly faster than exhaustive search, while still producing certified optimal solutions in practice.

Dual decomposition is a variant of *Lagrangian relaxation*, a general method for combinatorial optimization, and is also closely related to belief propagation for both graphical models and combinatorial structures. Lagrangian relaxation has been applied to a wide variety of problems in other domains as diverse as network routing and auction pricing. Recently, this technique has been used in natural language processing to solve decoding problems in combined parsing and tagging, non-projective dependency parsing, MT system combination, CCG supertagging, and syntactic translation.

2 Content Overview

This tutorial presents dual decomposition as a general inference technique, while utilizing examples and applications from natural language processing. The goal is for attendees to learn to derive algorithms for problems in their domain and understand the formal guarantees and practical considerations in using these algorithms.

We begin the tutorial with a step-by-step construction of an algorithm for combined CFG parsing and trigram tagging. To analyze this algorithm, we derive formal properties for the techniques used in the construction. We then give further examples for how to derive similar algorithms for other difficult tasks in NLP and discuss some of the practical considerations for using these algorithms on real-world problems. In the last section of the tutorial, we explore the relationship between these algorithms and the theory of linear programming, focusing particularly on the connection between linear programming and dynamic programming algorithms. We conclude by using this connection to linear programming to develop practical tightening techniques that help obtain exact solutions.

3 Outline

1. Step-By-Step Example Derivation
2. Formal Properties
3. Further Examples
   (a) Dependency Parsing
   (b) Translation Decoding
   (c) Document-level Tagging
4. Practical Considerations
5. Linear Programming Interpretation
6. Connections between DP and LP
7. Tightening Methods for Exact Solutions

Tutorial Abstracts of ACL/HLT 2011, page 6,
Portland, Oregon, USA, June 2011.

Author Index