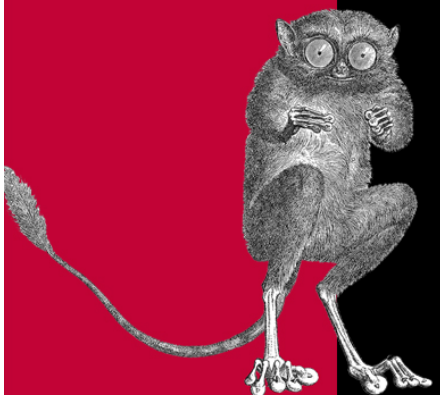
O'REILLY®

Developing Language Annotation for Machine Learning Algorithms

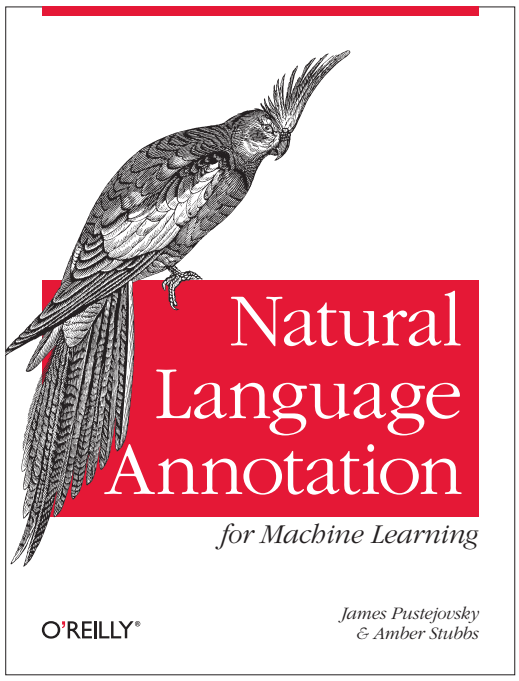
Brandeis
CS 140b

January 13, 2015



Professor Marie Meteer

Based on Brandeis CS 216 2011



Course Outline

- The Importance of Annotation
- Selecting an Annotation Task
- Model and Specification
- Annotation Essentials
- Overview of ML Algorithms
- Testing, Evaluation, and Revision
- Big Data

THE IMPORTANCE OF ANNOTATION

Natural Language Processing

- Applications that facilitate human interaction with machines and other devices through the use of natural language:
  - Machine Translation
  - Question Answering
  - Speech Recognition
  - Summarization
  - Document Classification

NLP and computers can do lots of things:

- book flights, map your road trip, play Chess, beat Ken Jennings at Jeopardy, …

BUT

- Human language is complex and dynamic; computers need to be "taught":
  - What words mean
  - How context changes meaning
  - When context is relevant
  - …

Annotation helps solve this problem

- Natural languages are not native to computer programs
- Search engines (for example) are effective, but they don't really understand language
- NLP Algorithms need intermediate data structures (annotation) to learn what people want from texts

What is annotation?

- Associating a label (metadata) with specific content in a document or file
- Annotation is everywhere
  - Image labeling
  - Spam detection
  - Date and event labeling
  - Document typing

Things you can do with annotation

- Document classification
- Named Entity Recognition
- Sentiment analysis
- Temporal reasoning
- Medical record processing

And so much more!

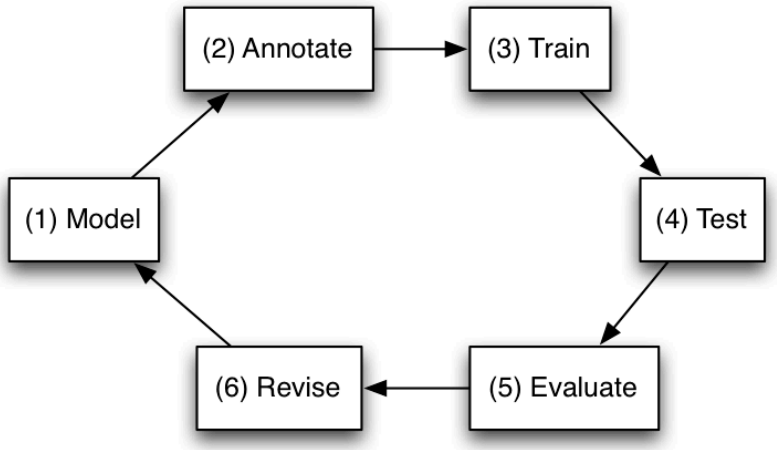
Aspects of Annotation

- Structure of language (word classes, syntax)
- Meaning of words (word sense disambiguation)
- Interpretation of meaning (semantics)
- Document structure (discourse)
- Speech sounds/parts of words (phonetics, morphology)

Creating annotations

- Interpreting language isn't simple; people don't always agree on meaning
- An annotated corpus needs to be consistent to train/test an algorithm
- Today we'll focus on the best practice for creating an annotated corpus, and how to use the corpus when you have it

The MATTER cycle



SELECTING AN ANNOTATION TASK

Where to begin?

- Natural language is complicated; computers can't understand it
- Annotation tasks need to be focused on a particular goal

Choosing a goal

- Goals can take lots of different forms
- Questions to answer:
  - What will the annotation be used for?
  - What will the overall outcome of the annotation be?
  - Where will the corpus come from?
  - How will the outcome be achieved?
- Think in terms of classification

Outcome of the Annotation

- What will the annotation be used for?
  - Database population, linguistic analysis, summarization, timeline creation…
- How in-depth will your classification be?

Where will the corpus come from?

- No stealing!
- Needs to be representative and balanced compared to your goal
- Size of the corpus
  - Trade-off between utility and practicality
- Look at existing corpora

Achieving the goal

- What aspects of the text will help you with your classification?
  - All the text in the document?
  - Individual words?
  - Relations between words? Between documents?

Things to consider

- Informativeness vs. correctness
- Scope of the task:
  - Styles/sources of text
  - Level of detail
- Purpose of the annotation
- Revisions will happen

MODEL AND SPECIFICATION

Goal → Model

- *Model = <T,R,I>*
  - Terms, relations, interpretations
- Terms = classifications applied to the data
  - Ex: “Spam”, “not-spam”, parts-of-speech
- Relations = connections between terms
  - Ex: link between a word and its definition
- Interpretation = what the metadata means
  - How to interpret the annotation

Example: Temporal Annotation

- Terms:
  - Timex3: tag used for temporal markers
    - (“today”, “October 3rd”, “last Monday”)
  - Event: tag used to mark occurrences
    - (“ran”, “voted”, “party”)
- Relations:
  - TLINK: connection between an Event and the Timex3 associated with its occurrence
- Interpretation:
  - An Event and Timex3 connected by a Tlink have a temporal connection that can be exploited

Model → Specification

- Specification = full description of the tags and attributes in your annotation
- Ex: TLINKs can indicate different temporal orderings:

```
<!ATTLIST TLINK relType (
   BEFORE | AFTER | INCLUDES | IS_INCLUDED | DURING |
DURING_INV  | SIMULTANEOUS | IAFTER | IBEFORE |
   IDENTITY | BEGINS | ENDS | BEGUN_BY            |
ENDED_BY ) #REQUIRED >
```


Time Stamping Events

April 25, 2010

- President Obama paid tribute Sunday to 29 workers killed in an explosion at a West Virginia coal mine earlier this month, saying they died "in pursuit of the American dream." The blast at the Upper Big Branch Mine was the worst U.S. mine disaster in nearly 40 years.Obama ordered a review earlier this month and blamed mine officials for lax regulation.

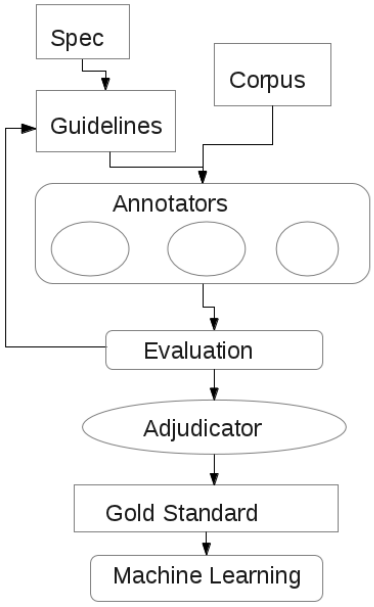
Temporal Ordering of Events

April 25, 2010

- President Obama paid tribute Sunday to 29 workers killed in an explosion at a West Virginia coal mine earlier this month, saying they died "in pursuit of the American dream." The blast at the Upper Big Branch Mine was the worst U.S. mine disaster in nearly 40 years.Obama ordered a review earlier this month and blamed mine officials for lax regulation.

ANNOTATION ESSENTIALS

Annotation Process



Choosing annotators

- What do your annotators need to know?
  - What language(s) do they need to speak?
  - Do they need to understand linguistics?
  - If your data is from a specialized field, do your annotators need another type of training?

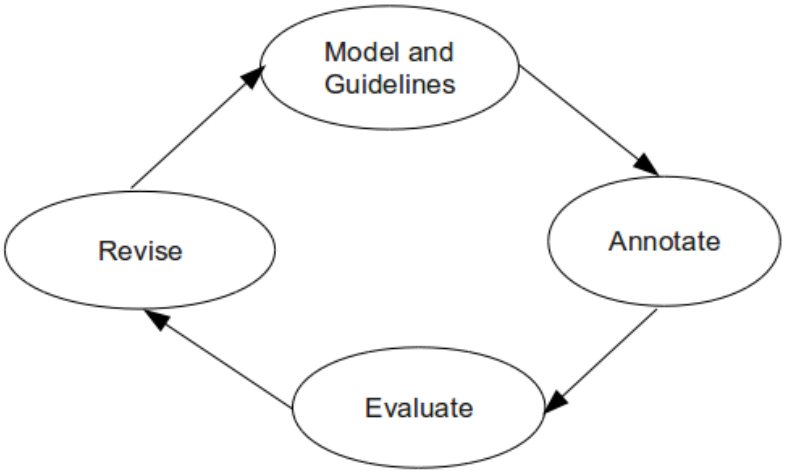
Specification → Guidelines

- Directions for the annotators
  - What, where, why, how
- Reusable
- Different from specification
  - Focus on how, why

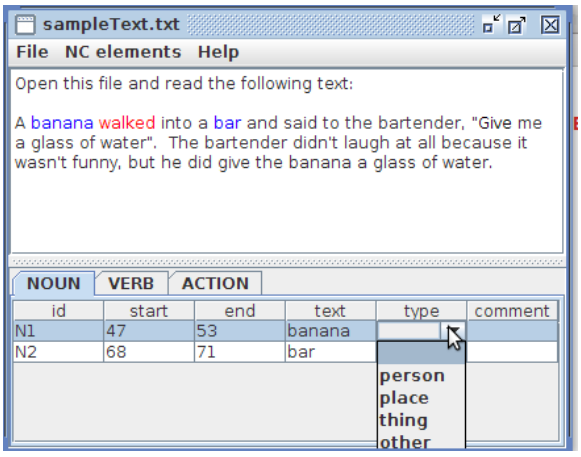
Guideline considerations

- What is the goal of the project?
- What is each tag called and how is it used?
- What parts of the text do you want annotated, and what should be left alone?
- How will the annotation be created?

MAMA cycle



Annotation Environment - MAE



Inter-annotator Agreement

- How clear/reproducible is your annotation task?
  - Could it be used by other people, or applied to another dataset?
- Kappa scores
  - Cohen (two annotators) and Fleiss (more than two)
  - What do they mean?

Adjudication

- Smooth over errors between annotators
  - Stay in line with the goal, spec, and guidelines
- Look for things that were missed
- End result: Gold Standard

OVERVIEW OF MACHINE LEARNING ALGORITHMS

What does an ML algorithm do?

- "Learning is any process by which a system improves its performance from experience."
  – Herbert Simon
- Annotation can give us a richer idea of what's in the dataset.
- We can leverage this knowledge as new features for training our algorithms.

Defining the learning task

Improving on a task, T, with respect to a performance metric, P, based on experience, E.

- Choose the “training experience”
- Identify the target function (what is the system going to learn?)
- Choose how to represent the target function
- Choose a learning algorithm
- Evaluate the results

Going from annotations to features

- Just because you have a lot of annotated data doesn’t mean your results are going to get better.

- Feature selection: the process of finding which features in your dataset are most helpful in solving your learning task.

Types of Features

- N-grams
  - classic “bag of words” approach to modeling a document. Each word in document is treated as a feature for learning algorithm.
- Structure-dependent
  - features that can be manipulated by virtue of the properties of the data structure itself.
- Annotation-dependent
  - any features associated with an annotation specification that reflects a *model* of the data.

Types of Learning

- Unsupervised learning
  - Finding structure from an input set of unlabeled data.
- Supervised learning
  - Generating a function that maps from inputs to a fixed set of labels.
- Semi-supervised learning
  - Generating a function that maps from inputs of both labeled and unlabeled data.

Unsupervised learning

- Requires no annotated data
- Clustering algorithms find regularities and hidden structure from the dataset
- Types of Clustering
  - Exclusive (e.g., k-means)
  - Overlapping
  - Hierarchical

Supervised Learning

- Goal is to train an algorithm with the most informative and representative examples you can find or create for your task.
- A classifier predicts what class or category a data element will belong to, after having trained it over a dataset. This requires that the data is labeled with particular classes (the labels you want on the data elements),with positive examples and negative examples for the desired classification.

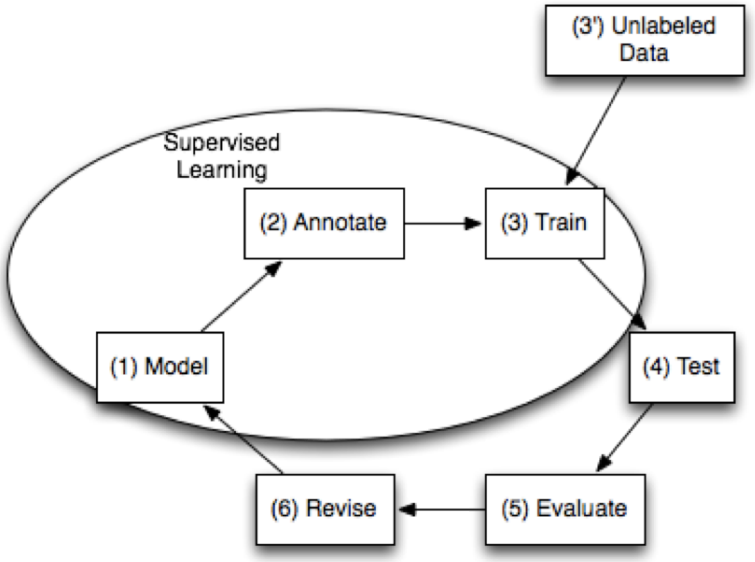
Supervised Learning Algorithms

- K-Nearest Neighbor
- Decision Tree
- Naïve Bayes
- Maximum Entropy (MaxEnt)
- Support Vector Machines
- Sequence Induction Algorithms
  - HMMs
  - CRFs
  - MEMMs

Semi-supervised learning

- A method of learning that employs both labeled data as well as unlabeled data.
- You may not have a rich and descriptive model with which to create an annotation of the dataset:
  - there are dependencies among elements in the data that are not identified.
- By first applying unsupervised learning technique, such as k-means, you might find clusters in data that reflect meaningful representations of complex or high dimensional data.
- The results of this step can then be taken as input for a supervised phase of learning, using these clusters as soft labels.

Semi-supervised learning



The "Train" can include:

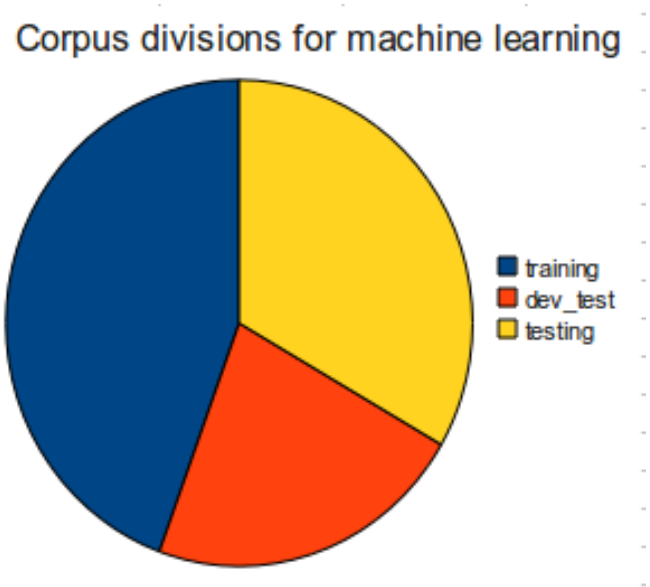
- Boosting:
  - Choose a strong learner from many weak ones
- Active Learning:
  - algorithm is allowed to choose data from which it learns
- Co-Training:
  - two "views" of the data being examined for learning, where each has an independent set of features
- Coupled-Training:
  - Links the simultaneous training of many extractors over data
- EM:
  - Estimates parameters of a generative model

Common uses for ML algorithms

Algorithm	Use
Naïve Bayes	Sentiment analysis, word sense disambiguation, event recognition, named entity identification
Support Vector Machines (SVMs)	Sentiment analysis, semantic roles, word sense disambiguation, semantic type
Decision Tree	Event recognition, co-reference resolution,
Conditional Random Field (CRF)	Part-of-speech tagging, named entity identification,
Maximum Entropy (MaxEnt)	Sentiment analysis, ontological class, event recognition

TESTING, EVALUATION, REVISION

Train-Test-Evaluate Cycle



Evaluation

- Compare your algorithm results to the Gold Standard
  - Precision
    - true positive /(true positive + false positive)
  - Recall
    - true positive / (true positive + false negative)
  - F-measure
    - Harmonic mean of precision and recall
- What do the scores mean?
  - Interpretation depends on the task

Revision

- What problems does your algorithm have?
  - Features?
  - Algorithm type?
  - Input data?
  - Model too specific/not specific enough?
- Fix your errors!

BIG DATA

Recap

- Goal
- Corpus
- Model → Specification
- Specification → Guidelines
- Guidelines → Annotation, Gold Standard
- Gold Standard → features, ML
- ML → Error detection and revision

Forward Looking

- Existing resources/standards
- Annotation environments
- Types of ML algorithms
- Crowdsourcing
- Other methods of data/annotation collection

Take-aways

- Annotation is an important part of using computers for processing natural languages
- The MATTER cycle provides a methodology for creating annotated corpora, regardless of the corpus medium or annotation goal
- Annotation projects can take time to do well, but the result is worth the effort

THANK YOU!

O'REILLY