Learning to Predict Structures
with Applications to Natural Language Processing

Ivan Titov
Before we start ...

- Fill in the survey:
  - Name
  - Email (Please make it eligible!)
  - Matriculation number
  - Department: CS or CoLi
  - BSc or MSc and your semester

- These two points will affect the set of topics and papers
  - Your major and research interests
  - Previous classes attended (or attending now)
    - Machine learning?
    - Statistical NLP?
    - Information Extraction?
Outline

- Introduction to the Topic
- Seminar Plan
- Requirements and Grading
Learning Machines to Do What?

This seminar is about supervised learning methods

1. Take a set of labeled examples \( \{(x_i, y_i)\}_{i=1}^{n} \), \( x \in \mathcal{X}, \ y \in \mathcal{Y} \)

2. Define a parameterized class of functions

\[ f(w) : \mathcal{X} \rightarrow \mathcal{Y} : w \in \mathbb{R}^n \]

- Represent images as vectors of features \( \varphi(x) \) (e.g., SIFT features for images)
- And consider linear functions:

\[ y = \arg\max_{y \in \mathcal{Y}} w_y \varphi(x) \]

Distinct vector for each \( y \): cats, dogs,..
Supervised Classification

- Linear functions:
  \[ y = \arg \max_{y \in \mathcal{Y}} w_y \varphi(x) \]

- We want to select “good” \( w \) such that it does not make mistake on new examples, i.e. for every \( x \) it predicts correct \( y^* \)
  \[ w_{y^*} \varphi(x) > \max_{y' \in \mathcal{Y}, y \neq y^*} w_{y'} \varphi(x) \]

- To do this we minimize some error measure on the finite training set
  - Having just a small error on the training set is not sufficient
  - For example, we may want it to be “confident” on the training set
    \[ w_{y^*} \varphi(x) - \max_{y' \in \mathcal{Y}, y \neq y^*} w_{y'} \varphi(x) > \gamma \]

- But what if \( y \in \mathcal{Y} \) is not a class label but a graph?
  - You cannot have an individual vector \( w_y \) for every \( y \)
Example Problems

- **Syntactic Parsing:**

  ![Example of syntactic parsing](image)

  *Bill sells fresh oranges*

- **3D Layout prediction for an image:**

  ![Example of 3D layout prediction](image)

- **Protein structure prediction (disulfide bonds):**

  ![Example of protein structure prediction](image)

  We cannot learn a distinct model for each $y$: we need to understand:
  - how to break it in parts
  - how to predict these parts
  - how these parts interact with each other
Skeleton of a general SP methods

- Decide how to represent your structures for learning

\[(x, y) = \]

\[\varphi(x, y) = (..., 1, 0, ..., 0, 1, ..., 1, ....)\]

- Note that we have \(\varphi(x, y)\) not \(\varphi(x)\) or \(\varphi(y)\)

\(x\) (input) is the sentence, \(y\) (output) is the tree
And then you define a vector \( w \), for example:

\[
(x, y^*) = (\text{John lost his pants})
\]

\[
\varphi(x, y^*) = (\ldots, 1, 0, \ldots, 0, 1, \ldots, 1, \ldots)
\]

\[
w = (\ldots, +5, +5, \ldots, -100, +2, \ldots, +3, \ldots)
\]

Their inner product is:

\[
w \varphi(x, y^*) = 5 \times 1 + 2 \times 1 + 3 \times 1 = 10
\]
Skeleton of a general SP methods

- What if we have some bad parse tree:

\[(x, y') = \begin{array}{c}
\text{John} \\
\text{lost} \\
\text{his} \\
\text{pants}
\end{array}
\]

\[\varphi(x, y') = (..., 1, 0, ..., 1, 0, ..., 1, ....)\]

\[w = (..., +5, +5, ..., -100, +2, ..., +3, ....)\]

- Their inner product is:

\[w\varphi(x, y') = 5 \times 1 - 100 \times 1 + 3 \times 1 = -92 \ll \varphi(x, y^*) = 10\]
Structured Prediction

- You use your model:
  \[ y = \arg \max_{y' \in \mathcal{Y}(x)} w \varphi(x, y') \]

- We want to select “good” \( w \) such that it does not make mistake on new examples, i.e. for every \( x \) it predicts correct \( y^* \)
  \[ w \varphi(x, y^*) > \max_{y' \in \mathcal{Y}(x), y \neq y^*} \varphi(x, y') \]

- To do this we again minimize some error measure on the finite training set
Structured Prediction (challenges)

1. Selecting feature representation $\varphi$
   - It should be sufficient to **discriminate correct trees from incorrect ones**
   - It should be possible to decode with it (see (3))

2. Learning
   - Which error function to optimize on the training set, for example
     $$w\varphi(x, y^*) - \max_{y' \in \mathcal{Y}(x), y \neq y^*} \varphi(x, y') > \gamma$$
   - How to make it efficient (see (3))

3. Decoding:
   $$y = \arg\max_{y' \in \mathcal{Y}(x)} w\varphi(x, y')$$
   - Dynamic programming for simpler representations $\varphi$?
   - Approximate search for more powerful ones?
Decoding: example

\[(x, y) = (\text{John}, \text{lost}, \text{his}, \text{pants})\]

\[\varphi(x, y) = (\ldots, 1, 0, \ldots, 0, 1, \ldots, 1, \ldots)\]

\[w = (\ldots, +5, +5, \ldots, -100, +2, \ldots, +3, \ldots)\]

Decoding: find the dependency tree which has the highest score

\[y = \arg\max_{y' \in \mathcal{Y}(x)} w\varphi(x, y')\]

Does it remind you something?
Decoding: example

- Select the highest scoring directed tree:

```
..., 1, 0, ..., 0, 1, ..., ...
... John lost Mary lost his his pants lost pants ...
```

- For this sentence only a subset is relevant and it can be represented as a weighted directed graph

```
..., +5, +5, ..., -100, +2, ..., +3, ...)
```

- Directed MST problem: Chi-Liu-Edmonds algorithm $O(n^2)$
Decoding: example

- What if we switch to slightly more powerful representation:
  - Counts of all subgraphs of size 3 instead of 2?

- This problem is NP-complete!
  - However, you can use approximations
  - Relaxations to find exact solutions in most cases
  - Consider non all subgraphs of size 3

- It is typical for structured prediction
  - Use a powerful model but approximate decoding or
  - A simpler model but exact search
Outline

- Introduction to the Topic
- Seminar Plan
- Requirements and Grading
Goals of the seminar

- Give an overview of state-of-the-art methods for structured prediction
  - If you encounter a structured prediction problem, you should be able to figure what to use and where to look
  - This knowledge is applicable outside natural language processing
- Learn interesting applications of the methods in NLP
- Improve your skills:
  - Giving talks
  - Presenting papers
  - Writing reviews
Plan

- Next class (November 5):
  - Introduction continued: Basic Structured Prediction Methods
    - Perceptron => Structured Perceptron
    - Naive Bayes => Hidden Markov Model
    - Comparison: HMM vs Structured Perceptron for Markov Networks
  - Decide on the paper to present (before Wednesday, November 2!)
    - On the basis of the survey and the number of registered students, I will adjust my list and it will be online on today
  - Starting from November 13: paper presentations by you
Topics (method-wise classification)

- **Hidden Markov Models vs Structured Perceptron**
  - Example: The same class of function but different learning methods (discriminative vs generative)

- **Probabilistic Context-Free Grammars (CFGs) vs Weighted CFGs**
  - Similar to above but for parsing (predicting trees)

- **Maximum-Entropy Markov models vs. Conditional Random Fields**
  - Talk about label-bias, compare with generative models,..

- **Local Methods vs Global Methods**
  - Example: Minimum Spanning Tree algorithm vs Shift-reduce parsing for dependency parsing
Topics (method-wise classification)

- Max-margin methods
  - Max-Margin Markov Networks vs Structured SVM

- Search-based models
  - Incremental perceptron vs SEARN

- Inference with Integer Linear Programming
  - Encoding non-local constraints about the structure of the outputs

- Inducing feature representations:
  - Latent-annotated PCFGs
    - Initial attempts vs split-merge methods
  - Incremental Sigmoid Belief Networks
    - ISBNs vs Max-Ent models
Topics (application-wise classification)

- **Sequence labelling type tasks:**
  - Part-of-speech tagging
    
    John lost his pants
    
    NNP VBD POS NNS

- ** Parsing tasks**
  - Phrase-tree structures
    
    
    
  - Dependency Structures
  - Semantic Role Labeling
    
    AGENCY
    THEME
    Sequa makes and repairs jet engines

- **Information extraction**
Requirements

- Present a talk to the class
  - In the most presentation you will need to cover 2 related papers and compare the approached
  - This way we will have (hopefully) more interesting talks
- Write 2 critical “reviews” of 2 selected papers (1-1.5 pages each)
  - Note: changed from 3!!
- A term paper (12-15 pages) for those getting 7 points
  - Make sure you are registered to the right “version” in HISPOS!
- Read papers and participate in discussion
  - If you do not read papers it will not work, and we will have a boring seminar
  - Do not hesitate to ask questions!
Grades

- **Class participation grade: 60 %**
  - You talk and discussion after your talk
  - Your participation in discussion of other talks
  - 2 reviews

- **Term paper grade: 40 %**
  - Only if you get 7 points, otherwise you do not need one
  - Term paper
Presentation

- Present the methods in accessible way
  - Do not (!) present something you do not understand
  - Do not dive into unimportant details

- Compare proposed methods
- Have a critical view on the paper: discuss shortcomings, any of ideas, any points you still do not understand (e.g., evaluation), any assumptions which seem wrong to you ...
- To give a good presentation in some cases you may need to read one or (maximum two) additional papers (e.g., those referenced in the paper)

- You can check the web for slides on a similar topics and use their ideas, but you should not reuse the slides

- See links to the tutorials on how to make a good presentation

- Send me your slides 1 week before the talk
  - I will give my feedback within 2 days of receiving
  - Often, we may need to meet and discuss the slides together
Term paper

- **Goal**
  - Describe the papers you presented in class
  - Your ideas, analysis, comparison (more later)
  - It should be written in a style of a research paper

- **Length:** 12 – 15 pages

- **Grading criteria**
  - Clarity
  - Paper organization
  - Technical correctness
  - New ideas are meaningful and interesting

- **Submitted in PDF to my email**
Critical review

- A short critical (!) essay reviewing papers presented in class
  - One paragraph presenting the essence of the paper (in your own words!)
  - Other parts underlying both positive sides of the paper (what you like) and its shortcomings

- The review should be submitted **before** its presentation in class
  - (Exception is the additional reviews submitted for the seminars you skipped, later about it)

- No copy-paste from the paper

- Length: 1-1.5 pages
Your ideas / analysis

- Comparison of the methods used in the paper with other material presented in the class or any other related work
- Any ideas on improvement of the approach
- ....
Attendance policy

- You can skip ONE class without any explanation
- Otherwise, you will need to write an additional critical review (for the paper which was presented while you were absent)
Office Hours

- I would be happy to see you and discuss after the talk from 16:00 – 17:00 on Fridays (may change if the seminar timing changes):
  - Office 3.22, C 7.4
- Otherwise, send me email and I find the time
  - Even preferable

- Please do meet me:
  - If you do not understand something (anything?)
    (Especially important if it is your presentation, but otherwise welcome too)
  - If you have suggestions and questions
Other stuff

- Timing of the class
- Survey (Doodle poll?)
- Select a topic to present and papers to review by Wed; November 2 (we will use Google docs)

- Note: earlier talks are easier...
  - We need a volunteer for November 12