Unsupervised Part-Of-Speech Tagging with Anchor Hidden Markov Models

Karl Stratos[1]

Joint work with Michael Collins[2] and Daniel Hsu[2]

[1]Bloomberg (work done while at Columbia University)

[2]Columbia University (Michael Collins is currently on leave at Google)

Unsupervised POS Tagging

- Quintessential unsupervised problem in NLP

`This/DET is/VERB unlabeled/ADJ text/NOUN`

- Naively estimating an HMM with the EM algorithm
  - Terrible performance!
  - Problem 1. **Model misspecification**
  - Problem 2. **Suboptimal learning**

- Extensions
  - Better models

    Hard-clustering HMM (Brown et al., 1992),
    Feature-rich models (Berg-Kirkpatrick et al., 2010)
  - Better learning

    Contrastive estimation (Smith and Eisner, 2005)
    Sparse prior (Johnson, 2007)

This Work

- **New model**: Anchor HMM
  - Each POS tag is "anchored" at some *unambiguous* word

NOUN	`loss`
ADP	`on`
NUM	`1`
DET	`the`

- **New learning algorithm**
  - Based on non-negative matrix factorization (Arora et al., 2012)
  - Exact, simple, and efficient

- Competitive with state-of-the-art on universal tagset

Overview

Anchor HMM

Learning Anchor HMM

- Non-Negative Matrix Factorization (NMF)
- Parameter Estimation

Experiments

Anchor HMM: Definition

- HMM with *structural restriction* on emission probabilities

$$p(x_1 \dots x_N, h_1 \dots h_N) = \pi(h_1) \times \prod_{i=1}^{N} o(x_i|h_i) \times \prod_{i=2}^{N} t(h_i|h_{i-1})$$

π : initial tag probabilities

o : emission probabilities

t : transition probabilities

Anchor HMM: Definition

- HMM with *structural restriction* on emission probabilities

$$p(x_1 \dots x_N, h_1 \dots h_N) = \pi(h_1) \times \prod_{i=1}^{N} o(x_i|h_i) \times \prod_{i=2}^{N} t(h_i|h_{i-1})$$

π : initial tag probabilities

o : emission probabilities

t : transition probabilities

- **Restriction**: each tag has at least 1 "anchor word" that belongs to that tag *only*.

$$o(\texttt{loss}|\text{NOUN}) = 0.0001 \qquad o(\texttt{loss}|h \neq \text{NOUN}) = 0$$

Anchor HMM: Definition

- HMM with *structural restriction* on emission probabilities

$$p(x_1 \dots x_N, h_1 \dots h_N) = \pi(h_1) \times \prod_{i=1}^{N} o(x_i|h_i) \times \prod_{i=2}^{N} t(h_i|h_{i-1})$$

π : initial tag probabilities

o : emission probabilities

t : transition probabilities

- **Restriction**: each tag has at least 1 "anchor word" that belongs to that tag *only*.

$$o(\texttt{loss}|\text{NOUN}) = 0.0001 \qquad o(\texttt{loss}|h \neq \text{NOUN}) = 0$$

- Reasonable assumption for POS tags

  True for all 10 languages in universal treebank (with 12 tags)

Game Plan

- Will exploit the anchor restriction to derive an exact parameter estimation algorithm.

- Key step: non-negative matrix factorization (NMF) of word-context co-occurrence matrix

Overview

Anchor HMM

Learning Anchor HMM

- Non-Negative Matrix Factorization (NMF)
- Parameter Estimation

Experiments

Context Representation

- $X \in \{1 \ldots n\}$: word
- $H \in \{1 \ldots m\}$: POS tag of X

Context Representation

- $X \in \{1 \ldots n\}$: word
- $H \in \{1 \ldots m\}$: POS tag of X
- Pick "context" representation $Y \in \mathbb{R}^d$ of X.
- Define matrix $\Omega \in \mathbb{R}^{n \times d}$ with rows $\Omega_x := \mathbf{E}[Y|X = x]$.

Context Representation

- $X \in \{1 \ldots n\}$: word
- $H \in \{1 \ldots m\}$: POS tag of X
- Pick "context" representation $Y \in \mathbb{R}^d$ of X.
- Define matrix $\Omega \in \mathbb{R}^{n \times d}$ with rows $\Omega_x := \mathbf{E}[Y|X = x]$.
- **Conditions on Y**
  1. Conditional independence

$$P(Y|X, H) = P(Y|H)$$

  2. Non-degeneracy

$$\text{rank}(\Omega) = m$$

Example Y

- Indicator vector of neighboring words $Y \in \{0,1\}^{2n}$

the dog saw the cat

1. $p(\texttt{dog}, \texttt{the} | \texttt{saw}, \textsc{verb}) = p(\texttt{dog}, \texttt{the} | \textsc{verb})$ ✓
2. $\Omega \in \mathbb{R}^{n \times 2n}$ has rank m. ✓*

*Unless the model is degenerate.

Factorization of Ω

- Under the conditions, $\Omega_x := \mathbf{E}[Y|X = x]$ factorizes:

$$\Omega_x = \sum_{h=1}^{m} p(h|x) \times \mathbf{E}[Y|h]$$

Factorization of Ω

- Under the conditions, $\Omega_x := \mathbf{E}[Y|X = x]$ factorizes:

$$\Omega_x = \sum_{h=1}^{m} p(h|x) \times \mathbf{E}[Y|h]$$

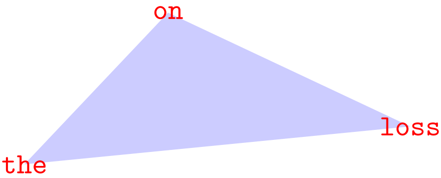
- **If x is an anchor**: $\Omega_x = \mathbf{E}[Y|h_x]$

Factorization of Ω

- Under the conditions, $\Omega_x := \mathbf{E}[Y|X = x]$ factorizes:

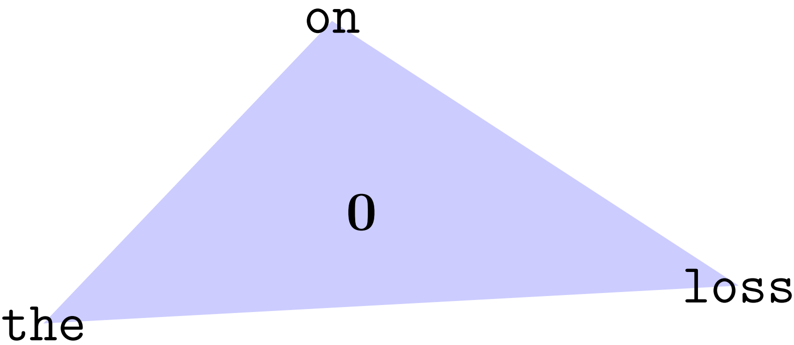
$$\Omega_x = \sum_{h=1}^{m} p(h|x) \times \mathbf{E}[Y|h]$$

- **If x is an anchor**: $\Omega_x = \mathbf{E}[Y|h_x]$

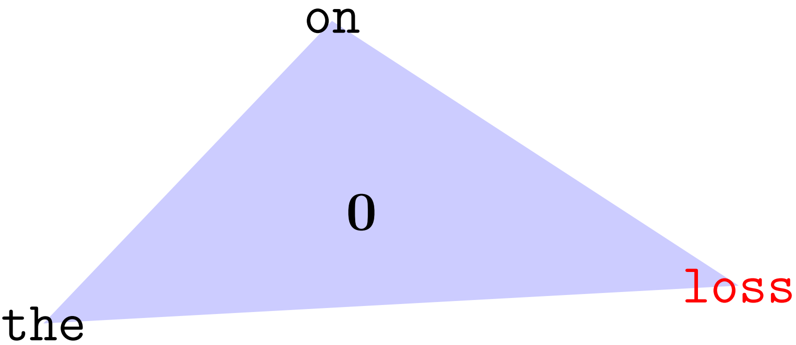


Ω_x form a **convex hull** with anchor words at m vertices.

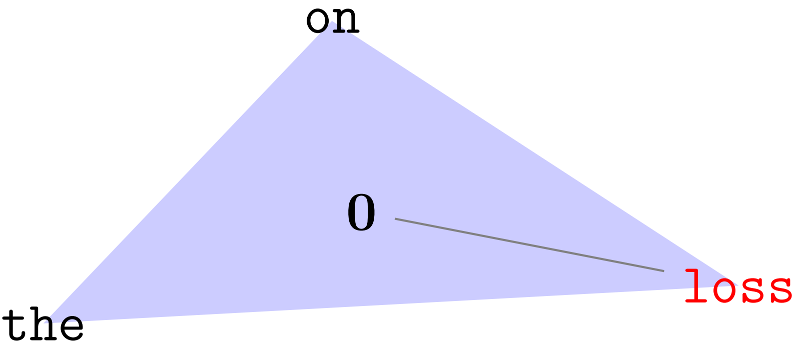
Finding Anchors (Arora et al., 2012)



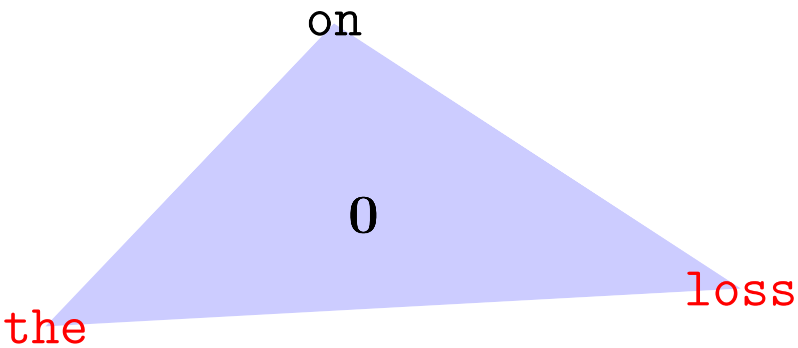
Finding Anchors (Arora et al., 2012)



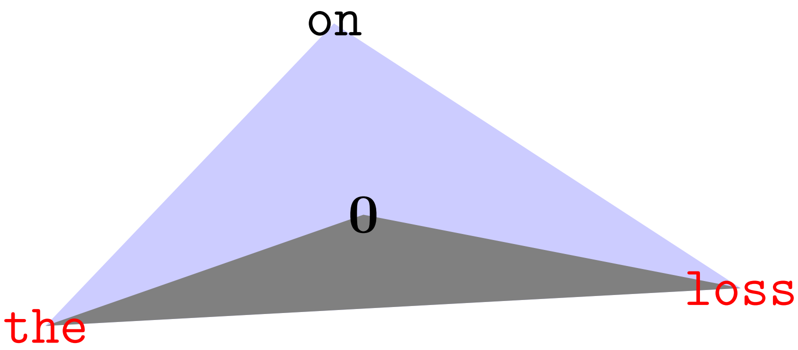
Finding Anchors (Arora et al., 2012)



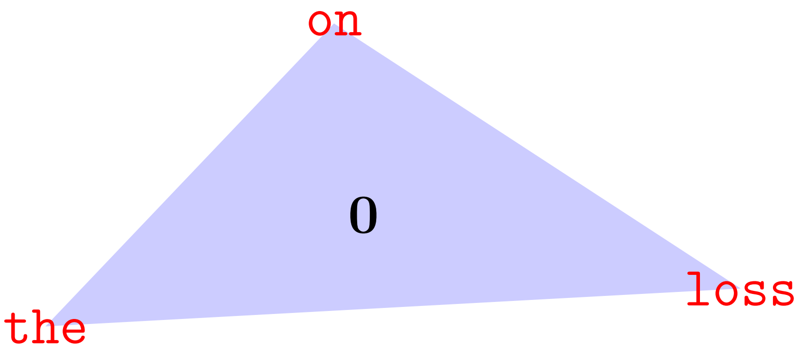
Finding Anchors (Arora et al., 2012)



Finding Anchors (Arora et al., 2012)



Finding Anchors (Arora et al., 2012)



NMF

Input. Ω with rows $\Omega_x = \mathbf{E}[Y|X=x]$, number of anchors m
Conditions. $Y \perp\!\!\!\perp X \mid H$, $\text{rank}(\Omega) = m$

NMF

Input. Ω with rows $\Omega_x = \mathbf{E}[Y|X=x]$, number of anchors m
Conditions. $Y \perp\!\!\!\perp X \mid H$, $\text{rank}(\Omega) = m$

1. Find anchor rows $\Omega_{a_1} \ldots \Omega_{a_m}$.

NMF

Input. Ω with rows $\Omega_x = \mathbf{E}[Y|X = x]$, number of anchors m
Conditions. $Y \perp\!\!\!\perp X \mid H$, $\text{rank}(\Omega) = m$

1. Find anchor rows $\Omega_{a_1} \ldots \Omega_{a_m}$.

2. Express each row Ω_x as a convex combination of anchor rows:

$$\Omega_x = \sum_{h=1}^{m} p(h|x) \times \Omega_{a_h}$$

   Can be solved with Frank-Wolfe.

NMF

Input. Ω with rows $\Omega_x = \mathbf{E}[Y|X = x]$, number of anchors m
Conditions. $Y \perp\!\!\!\perp X \mid H$, $\text{rank}(\Omega) = m$

1. Find anchor rows $\Omega_{a_1} \dots \Omega_{a_m}$.

2. Express each row Ω_x as a convex combination of anchor rows:

$$\Omega_x = \sum_{h=1}^{m} p(h|x) \times \Omega_{a_h}$$

   Can be solved with Frank-Wolfe.

Output. $p(h|x)$ for all tags h, words x

Overview

Anchor HMM

Learning Anchor HMM

- Non-Negative Matrix Factorization (NMF)
- Parameter Estimation

Experiments

Basic Idea

- $\Omega_x = \mathbf{E}[Y|X = x]$ can be estimated from unlabeled data.

- NMF of Ω gives "flipped" emission probabilities $p(h|x)$.
  - Use them to solve for model parameters.

Algorithm

1. Estimate $\widehat{\Omega}$ by counting word-context cooccurrences:

$$[\widehat{\Omega}_x]_i = \hat{p}(y_i|x) = \frac{\mathsf{count}(x, y_i)}{\mathsf{count}(x)}$$

2. Compute $\hat{p}(h|x) \leftarrow \mathsf{NMF}(\widehat{\Omega}, m)$.

3. Use Bayes' rule to recover emission parameters

$$\hat{o}(x|h) \leftarrow \frac{\hat{p}(h|x) \times \hat{p}(x)}{\sum_{x=1}^{n} \hat{p}(h|x) \times \hat{p}(x)}$$

4. Given $\hat{o}$, recover $\hat{t}$ and $\hat{\pi}$ (easy).

Practical Issues: Dimensionality Reduction

- Context $Y \in \mathbb{R}^{2n}$ is sparse and high-dimensional.
  - Cumbersome to work with.

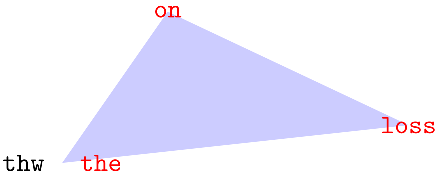
- Can use projection $\Pi \in \mathbb{R}^{2n \times d}$ to reduce dimension
  - Conditional independence does not break: $Y\Pi \perp\!\!\!\perp X \mid H$
  - Must ensure that $\Omega\Pi$ has rank m.

- Various choices of Π:
  - Random projection (Arora et al., 2012)
  - Projection onto best-fit subspace (i.e., SVD)
  - Projection based on canonical correlation analysis (CCA)
  - Projection based on hard-clustering assumption

Practical Issues: Better Anchors

- **Issue.** Anchors tend to be extremely rare words

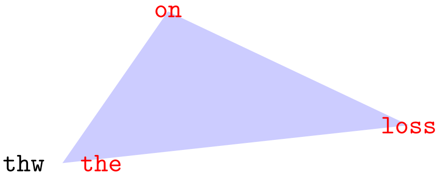
  Fix. Only consider top K frequent words as anchor candidates



Practical Issues: Better Anchors

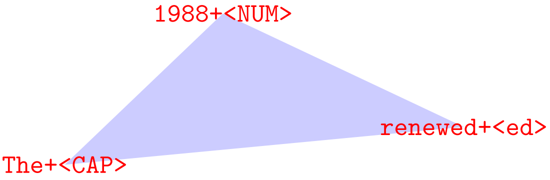
- **Issue.** Anchors tend to be extremely rare words

  Fix. Only consider top K frequent words as anchor candidates



- **Issue.** No spelling information used

  Fix. Augment Ω_x with spelling features



Overview

Setting

- **Dataset.** Universal treebank (McDonald et al., 2013)
  12 POS tags for 10 languages
  Hyperparameters tuned on English portion

- All models trained with 12 hidden states and evaluated on many-to-1 accuracy

- **Models.**
  - EM: HMM trained with EM
  - BROWN: Brown clusters (Brown et al., 1993)
  - ANCHOR: Anchor HMM
  - ANCHOR-FEAT: Anchor HMM + spelling features
  - LOG-LINEAR: Log-linear model with same features (Berg-Kirkpatrick et al., 2010)

Context for Learning Anchor HMM

- $Y \in \mathbb{R}^{2n}$: previous and next words

the dog saw the cat

- Choice of dimensionality reduction

	Accuracy (English)
Random	48.2
Best-Fit	53.4
CCA	57.0
Hard	**66.1**

Results: 12 Universal Tags

	de	en	es	fr	id	it	ja	ko	pt-br	sv
EM	46	60	61	60	50	52	60	52	60	42
BROWN	60	63	67	66	59	**66**	60	48	**67**	**62**
ANCHOR	61	66	69	68	64	60	65	54	65	51
ANCHOR-FEAT	63	**71**	**74**	**72**	**67**	60	69	**62**	66	61
LOG-LINEAR	**68**	62	67	62	61	53	**78**	61	63	57

- Anchor HMM: generally good performance
  - Spelling features help.

Results: 45 Original Tags (English)

	Accuracy
EM	62.6 (1.1)
CLUSTER	65.6
ANCHOR	67.2
ANCHOR-FEAT	67.7
LOG-LINEAR	**74.9** (1.5)

- Behind LOG-LINEAR
- Possible reason: spelling features more important with fine-grained tags

Discovered Anchor Words (for 12 Tags)

German	English	Spanish	French	Italian	Korean
empfehlen	loss	y	avait	radar	완전
wie	1	hizo	commune	però	중에
;	on	-	Le	sulle	경우
Sein	one	especie	de	-	줄
Berlin	closed	Además	président	Stati	같아요
und	are	el	qui	Lo	많은
,	take	países	(	legge	,
-	,	la	à	al	볼
der	vice	España	États	far-	자신의
im	to	en	Unis	di	받고
des	York	de	Cette	la	맛있는
Region	Japan	municipio	quelques	art.	위한

- `loss` ≈ noun   `1` ≈ number   `on` ≈ preposition   . . .
- Not perfect, but reasonable

Summary

- New model & algorithm for unsupervised POS tagging
  - **Anchor HMM**: each tag "anchored" at unambiguous word
  - **NMF-based learning**: exact, simple, and efficient
- Automatically discovers anchor words
  - Interpretable model

h_1	`loss`
h_2	`on`
h_3	`1`
h_4	`the`

- Future directions
  - Can exploit anchor assumption to learn richer model families?
  - Can we relax the anchor assumption further?

EXTRA SLIDES

Relation to Other HMM Variants

- HMM emission probabilities in matrix form O

$$O_{x,h} := o(x|h)$$

$$\begin{array}{cc} & \begin{array}{cc} \text{N} & \text{V} \end{array} \\ \begin{array}{c} \texttt{loss} \\ \texttt{set} \\ \texttt{hit} \\ \texttt{ran} \end{array} & \begin{bmatrix} 0.4 & 0.1 \\ 0.3 & 0.2 \\ 0.2 & 0.3 \\ 0.1 & 0.4 \end{bmatrix} \end{array}$$

General HMM

$$\begin{array}{cc} & \begin{array}{cc} \text{N} & \text{V} \end{array} \\ \begin{array}{c} \texttt{loss} \\ \texttt{set} \\ \texttt{hit} \\ \texttt{ran} \end{array} & \begin{bmatrix} 0.6 & 0.0 \\ 0.4 & 0.0 \\ 0.0 & 0.4 \\ 0.0 & 0.6 \end{bmatrix} \end{array}$$

Hard-clustering HMM
(Brown et al., 1992)

$$\begin{array}{cc} & \begin{array}{cc} \text{N} & \text{V} \end{array} \\ \begin{array}{c} \texttt{loss} \\ \texttt{set} \\ \texttt{hit} \\ \texttt{ran} \end{array} & \begin{bmatrix} 0.4 & 0.0 \\ 0.3 & 0.2 \\ 0.2 & 0.3 \\ 0.0 & 0.4 \end{bmatrix} \end{array}$$

Anchor HMM