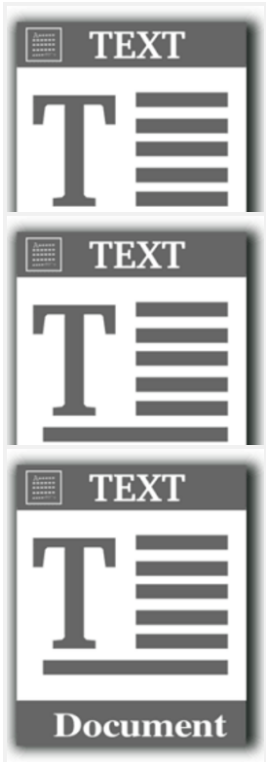
Leveraging Knowledge and Constraints in NLP and Sentiment Analysis

黄民烈
aihuang@tsinghua.edu.cn
清华大学计算机系
人工智能实验室

How to Build a Statistical Learning System for NLP?



Learning Machine

Desired output

Option 1: Supervised Learning

This approach does not scale to every task and domain

have: unlabeled data

hire: $$$$$$$$$$$$

linguist

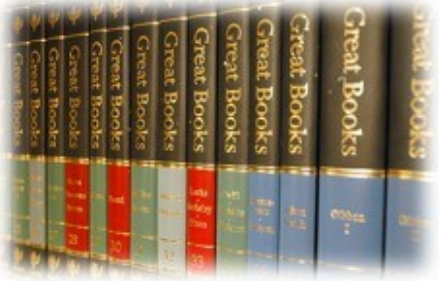
annotators

Option II: Unsupervised Learning

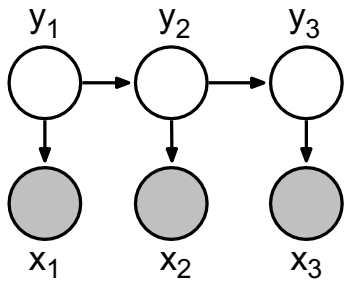
The true generative process is typically:

- **unknown**
- **complex; hard to model efficiently**

have: unlabeled data



design: model



train:

to maximize likelihood of observed data

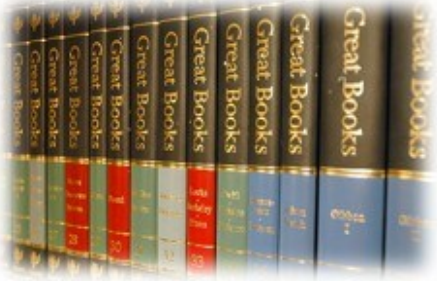
Option II: Unsupervised Learning

The true generative process is typically:

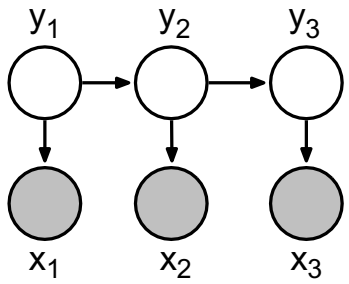
- **unknown**
  - **complex; hard to model efficiently**

<u>Result:</u> maximizing likelihood may not give expected output ...

have: unlabeled data



design: model



train:

to maximize likelihood of observed data

Option III: Semi-supervised Learning

Expectation Maximization:
may not be robust over unlabeled data; does not perform consistently better than the counterpart model

Chawla and Karakoulas 2005. JAIR
Cozman and Cohen, 2006
Mann and Andrew, ICML 2007
Su et al. ICML 2011

Have:

Unlabeled data

Few labeled data

Train:

to maximize likelihood of observed data

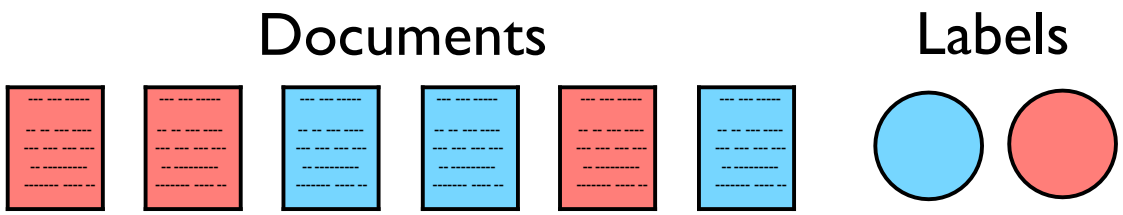
$$\max_{\theta} \sum_{d \in L} logP(c,d) + \sum_{d \in U} logP(d)$$

Prior Knowledge & Constraints

We posses a wealth of prior knowledge about most NLP tasks

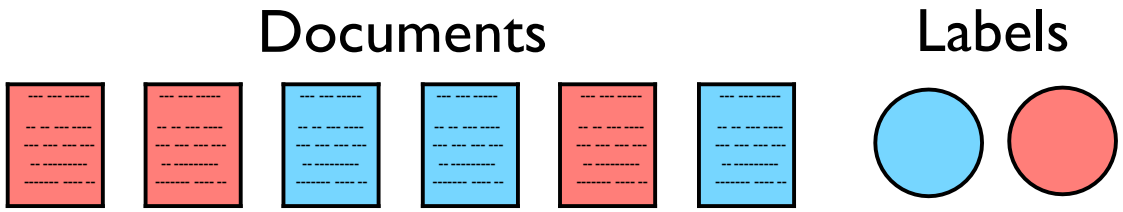
Knowledge can be formulated into constraints and many others math formulas

Example: Text Classification



- **Prior Knowledge:**
  - labeled features: information about the labels for documents that contain a particular word ***w***

Example: Text Classification



- **Prior Knowledge:**
  - labeled features: information about the labels for documents that contain a particular word ***w***

sentiment polarity

positive	negative
memorable	terrible
perfect	boring
exciting	mess

newsgroups classification

baseball	Mac	politics	...
hit	Apple	senate	...
Braves	Macintosh	taxes	...
runs	Powerbook	liberal	...

Example: Information Extraction

Extraction from research papers:

W.H. Enright. Improving the efficiency of matrix operations in the numerical solution of stiff ordinary differential equations. ACM Trans. Math. Softw., 4(2), 127-136, June 1978.

Prior Knowledge:

- labeled features:
  - *the word **ACM** should be labeled either **journal** or **conference** most of the time*

Example: Information Extraction

Extraction from research papers:

W.H. Enright. Improving the efficiency of matrix operations in the numerical solution of stiff ordinary differential equations. ACM Trans. Math. Softw., 4(2), 127-136, June 1978.

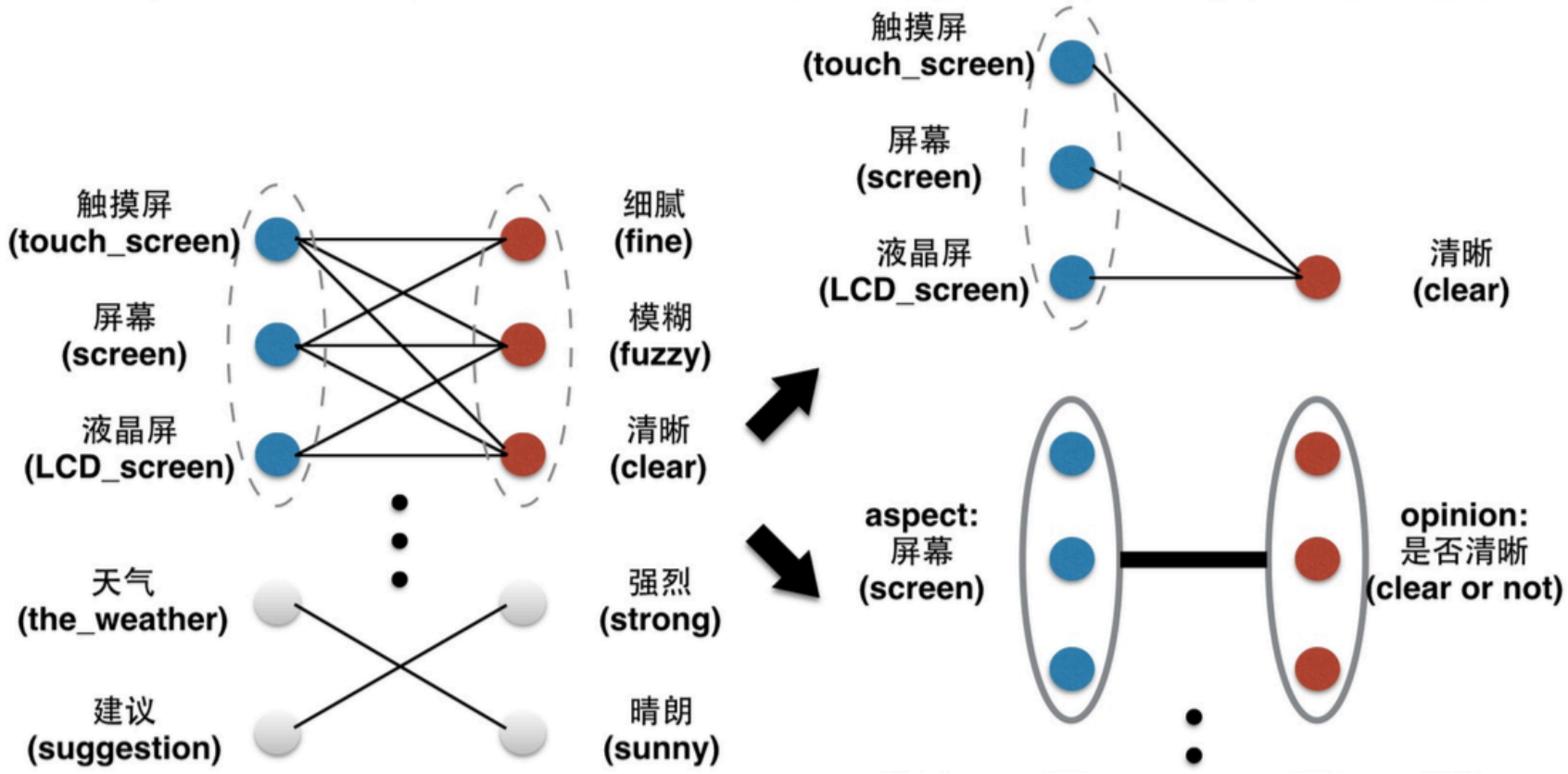
Prior Knowledge:

- labeled features:
  - *the word **ACM** should be labeled either **journal** or **conference** most of the time*
- non-Markovian (long-range) dependencies:
  - *each reference has at most one segment of each type*

Example: Sentiment Extraction

Prior Knowledge:

- ***Specific terms (e.g. high resolution) can only modify few particular aspects (e.g. screen)***



Example: Sentiment Extraction

Prior Knowledge:

- ***We can easily write a few signature terms for each aspect when performing Aspect detection***

daisy_bo VIP

口味：4 环境：4 服务：4 人均：120

生日的时候跟朋友来吃的井格，念念不忘，所以又拉着同事过来啦。这家店就在望京新世界一层侧面，招牌很显眼，很好找。喜欢店里的香油，都是直接一瓶一瓶的，倒起来那叫一个爽！巴蜀牛肉非常嫩爽，能吃辣的建议吃麻辣牛肉哇！有小麻花，但是没有烧饼和油条，南方人表示涮油条简直好吃到飞，希望可以上！另外，才知道原来井格就是以前的宽板凳啊……井格味道更好一点，嘻嘻

收起

墨 2016-05-07 829 有用

如果说你真的要走 / 把你的盾牌还给我 / 在你身上也没有用 / 我可以还给我爸爸

Jaqen H'ghar 2016-04-30 1178 有用

完成度相当高，政治大格局虽有点弱，但是情节合理紧凑，节奏恰当，小蜘蛛太太太亮眼！蚁人奥特曼也好赞！！但有两点：1. 黑寡妇,agent 13人设略崩，marvel还是处理女性角色无能。2. 你爸死了没关系我老婆还活着，你爸妈死了没关系我老婆还活着，我老婆大过天，我要跟我老婆在一起，人挡杀人佛挡杀佛

dr13 2016-04-30 1216 有用

没看出什么有意义的价值讨论，就是两边都很固执骄傲的一群人。对比之下黑豹才像个真正的领袖，而幻视则是在铁人队已经全线降低战斗逼格的情况下全程思考人生尽力旁观不开挂的哲学家。漫威架构大了后不连贯也开始明显，剧里为队长自责一生不要命的开飞机去找他的霍华德让人很难认同片中冷漠无情的队长。

Price:{价格，花费，便宜，...}

Service:{服务，服务员， ... }

Favor:{口味，味道， ... }

Storyline:
{“故事”,“情节”,“题材”,“剧本”,“编剧” }
Music:
{“主题曲”,“片尾曲”,“歌” }

Notation & Models

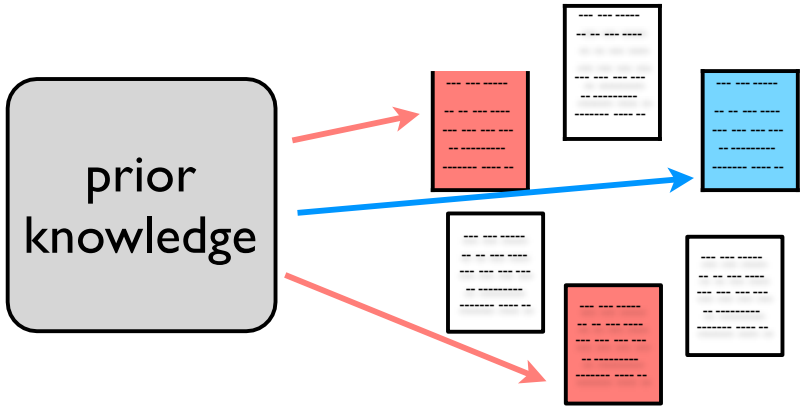
input variables (documents, sentences):	$\mathbf{x}$
structured output variables (parses, sequences):	$\mathbf{y}$
unstructured output variables (labels):	y
input / output variables for entire corpus:	$\mathbf{X}\ \mathbf{Y}$
probabilistic model parameters:	θ
generative models:	$p_\theta(\mathbf{x}, \mathbf{y})$
discriminative models:	$p_\theta(\mathbf{y}\|\mathbf{x})$
model feature function:	$\mathbf{f}(\mathbf{x}, \mathbf{y})$

Leveraging Prior Knowledge & Constraints

Possible approaches and their limitations.

Limited Approach: Labeling Data

approach: *Use prior knowledge to label data.*



Prototypes (+ cluster features):

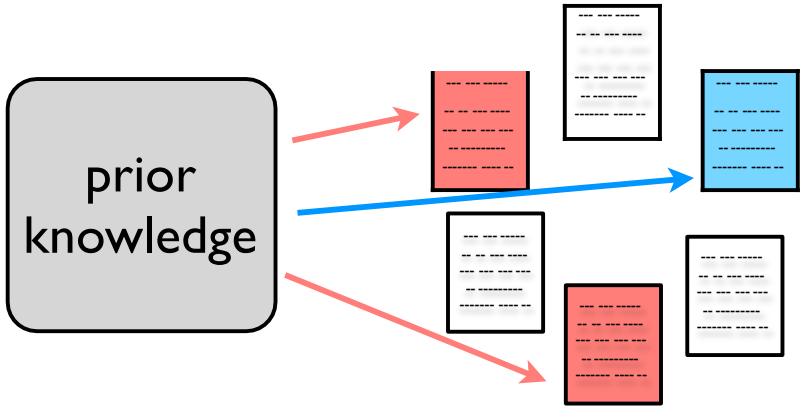
- [Haghighi & Klein 06]

Others:

- [Raghavan & Allan 07]
- [Schapire et al. 02]

Limited Approach: Labeling Data

approach: *Use prior knowledge to label data.*



Prototypes (+ cluster features):

- [Haghighi & Klein 06]

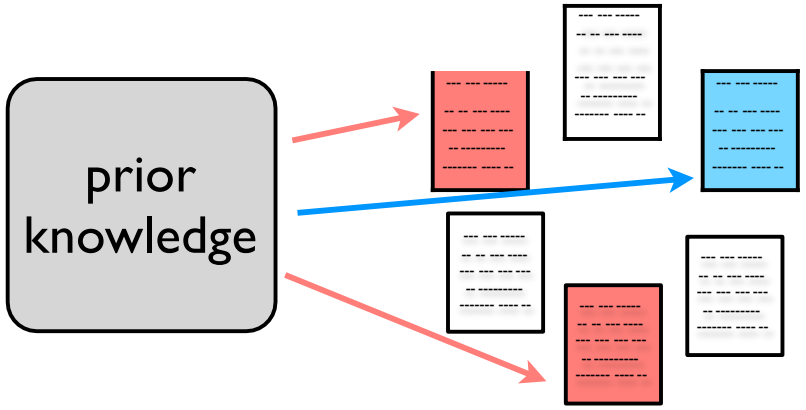
Others:

- [Raghavan & Allan 07]
- [Schapire et al. 02]

limitation: Often unclear how to label data.

Limited Approach: Labeling Data

approach: *Use prior knowledge to label data.*



Prototypes (+ cluster features):

- [Haghighi & Klein 06]

Others:

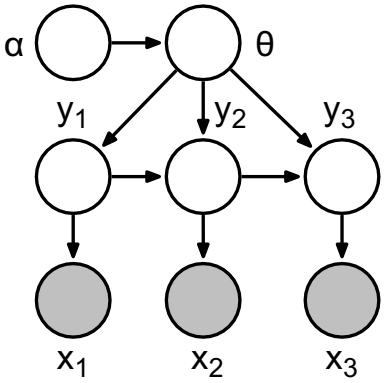
- [Raghavan & Allan 07]
- [Schapire et al. 02]

limitation: Often unclear how to label data.

- **Example #:** often (not always) *game* → {hockey,baseball}

Limited Approach: Bayesian Approach

approach: *Encode prior knowledge with a prior on parameters.*



specifying $p(\theta)$

natural: "θ should be *small* (or *sparse*)"

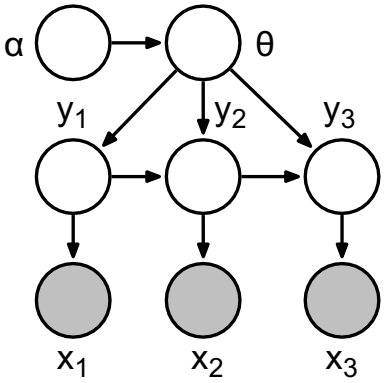
[Johnson 07], among many others

possible: "θ_i should be close to $\tilde{\theta}_i$"

(informative prior) [Dayanik et al. 06]

Limited Approach: Bayesian Approach

approach: *Encode prior knowledge with a prior on parameters.*



specifying $p(\theta)$

natural: "θ should be *small* (or *sparse*)"

[Johnson 07], among many others

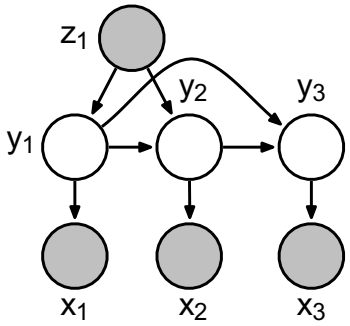
possible: "θ_i should be close to $\tilde{\theta}_i$"

(informative prior) [Dayanik et al. 06]

limitation: Our prior knowledge is not about parameters!
Parameters are difficult to interpret;
Hard to get desired effect.

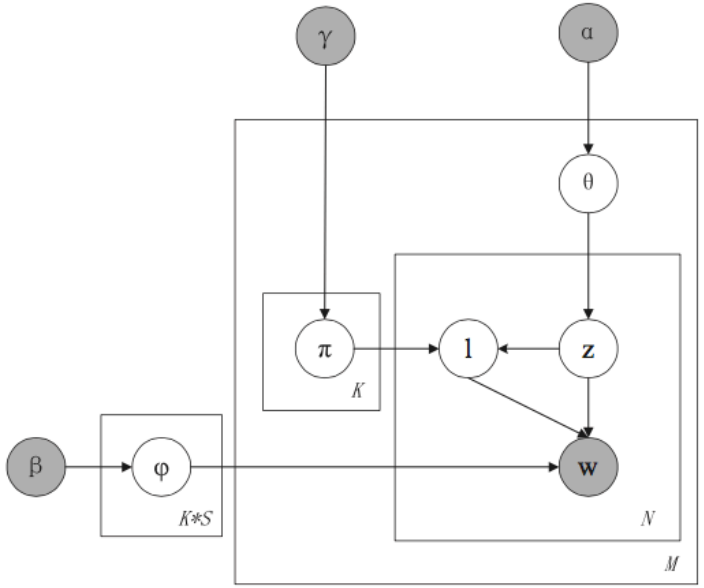
Limited Approach: Augmenting Model

approach: *Encode prior knowledge with additional variables and dependencies.*



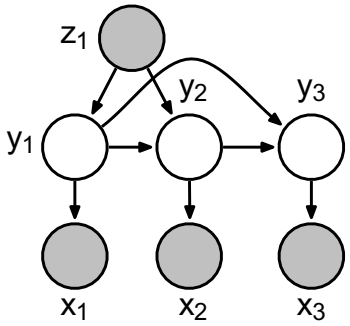
[Li 2009], (arguably) many unsupervised methods

[Li, Huang, Zhu 2010], modeling discourse relations with additional variables



Limited Approach: Augmenting Model

approach: *Encode prior knowledge with additional variables and dependencies.*

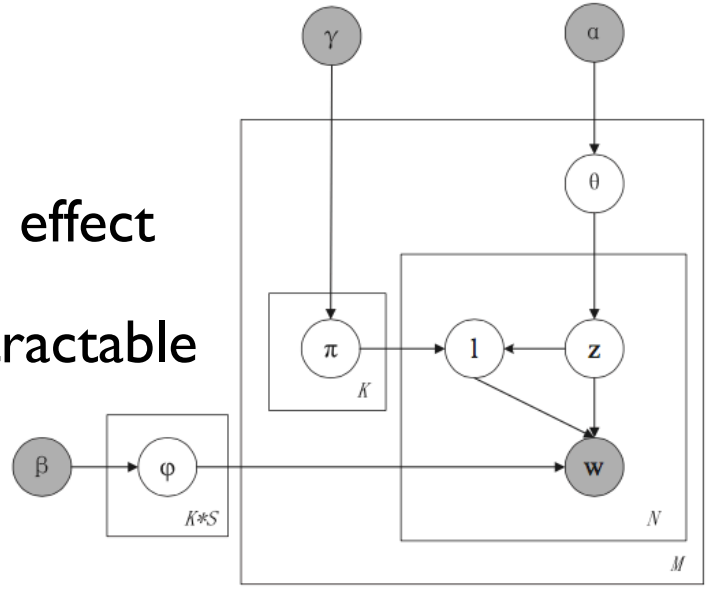


[Li 2009], (arguably) many unsupervised methods

[Li, Huang, Zhu 2010], modeling discourse relations with additional variables

limitation: can be difficult to get desired effect

limitation: may make exact inference intractable



How can we address these limitations?

Constraint Features & Expectations: Text Classification

- **constraint feature:**

$$\phi_{w\ell}(\mathbf{x}, y) = \begin{cases} 1 & \text{if } game \text{ is in } \mathbf{x} \text{ and } y \text{ is } hockey \\ 0 & \text{otherwise} \end{cases}$$

- **expectation:** $\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] = \frac{1}{c_w} \sum_{\mathbf{x}} \sum_{y} p_\theta(y|\mathbf{x}) \phi_{w\ell}(\mathbf{x}, y)$

Constraint Features & Expectations: Text Classification

- **constraint feature:**

$$\phi_{w\ell}(\mathbf{x}, y) = \begin{cases} 1 & \text{if } game \text{ is in } \mathbf{x} \text{ and } y \text{ is } hockey \\ 0 & \text{otherwise} \end{cases}$$

- **expectation:** $\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] = \frac{1}{c_w} \sum_{\mathbf{x}} \sum_{y} p_\theta(y|\mathbf{x}) \phi_{w\ell}(\mathbf{x}, y)$

  - ***expected*** probability that documents that contain *game* are labeled hockey (c_w is the count of *game*)

Constraint Features & Expectations: Text Classification

- **constraint feature:**

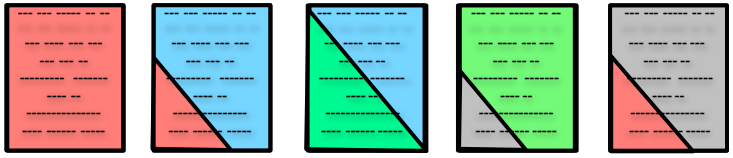
$$\phi_{w\ell}(\mathbf{x}, y) = \begin{cases} 1 & \text{if } \textit{game} \text{ is in } \mathbf{x} \text{ and } y \text{ is } \textit{hockey} \\ 0 & \text{otherwise} \end{cases}$$

- **expectation:** $\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] = \frac{1}{c_w} \sum_{\mathbf{x}} \sum_{y} p_\theta(y|\mathbf{x}) \phi_{w\ell}(\mathbf{x}, y)$

  - ***expected*** probability that documents that contain *game* are labeled hockey (c_w is the count of *game*)

labels
baseball
hockey
politics
science

contain *game*



Constraint Features & Expectations: Text Classification

- **constraint feature:**

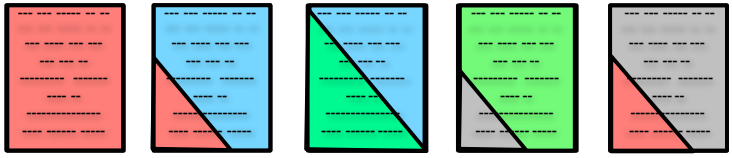
$$\phi_{w\ell}(\mathbf{x}, y) = \begin{cases} 1 & \text{if } \textit{game} \text{ is in } \mathbf{x} \text{ and } y \text{ is } \textit{hockey} \\ 0 & \text{otherwise} \end{cases}$$

- **expectation:** $\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] = \frac{1}{c_w} \sum_{\mathbf{x}} \sum_{y} p_\theta(y|\mathbf{x}) \phi_{w\ell}(\mathbf{x}, y)$

  - ***expected*** probability that documents that contain *game* are labeled hockey (c_w is the count of *game*)

labels
baseball
hockey
politics
science

contain *game*



(0.0 + 0.7 + 0.5 + 0.0 + 0.0) / 3 = 0.4

Constraining Model Expectations

- express preferences using **target values**: $\mathbf{b}$

- **Example #1 Constraint:** $\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] \approx \mathbf{b}$
  - *label distribution* for *game* is close to [40% 40% 20%]

Constraining Model Expectations

- express preferences using **target values**: $\mathbf{b}$

- **Example #1 Constraint:** $\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] \approx \mathbf{b}$
  - *label distribution* for *game* is close to [40% 40% 20%]

- **Example #2 Constraint:** $\mathbf{E}_{p_\theta}[\phi_m(\mathbf{x}, \mathbf{y})] \leq \mathbf{b}$
  - *expected* number of target words that align with *animada* is at most **1**

Some Related Frameworks

- PR: Posterior Regularization
- CODL: Constraint Driven Learning
- GEC: Generalized Expectation Criterion

Overview: PR, CoDL, GE

Constraint Driven Learning:

Apply constraints at decode time + self-training.

$$\arg\max_{\mathbf{Y}} \quad \log p_\theta(\mathbf{Y}|\mathbf{X}) - \text{penalty}(\mathbf{Y})$$

Generalized Expectation Constraints:

Train model to satisfy constraints.

$$\max_\theta \quad \mathcal{L}_\theta \quad \Longrightarrow \quad \max_\theta \quad \mathcal{L}_\theta - \text{penalty}(p_\theta(\mathbf{Y}|\mathbf{X}))$$

Posterior Regularization:

Project onto a constraint set + EM training.

$$\max_\theta \quad \mathcal{L}_\theta \quad \Longrightarrow \quad \max_\theta \quad \mathcal{L}(\theta; D_L) \; - \; \mathcal{D}_{\text{KL}}(\mathcal{Q} || \; p_\theta(\mathbf{Y}|\mathbf{X}))$$

For concreteness: running example

Want to ensure that 25% of unlabeled documents are about politics

- *constraint* features

$$\phi(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \text{if } \mathbf{y} \text{ is "politics"} \\ 0 & \text{otherwise} \end{cases}$$

- preferred expected value

$$\mathbf{b} = 0.25$$

- Expectation w.r.t. unlabeled data

$$\mathbf{E}_{p_\theta}[\phi_{w\ell}(\mathbf{X}, \mathbf{Y})] = \frac{1}{c_w} \sum_{\mathbf{x}} \sum_{y} p_\theta(y|\mathbf{x}) \phi_{w\ell}(\mathbf{x}, y)$$

Posterior Regularization

Posterior Regularization

J. Graça, K. Ganchev, B. Taskar (2007).

University of Pennsylvania (2007)

Idea: Regularize data likelihood with *valid* posteriors

$$\mathcal{L}(\theta) = \log p_\theta(\mathbf{X}_L, \mathbf{Y}_L) + \log \sum_{\mathbf{Y}} p_\theta(\mathbf{X}, \mathbf{Y}) + \log p(\theta)$$

$$\max_\theta \quad \mathcal{L}(\theta) \; - \; \min_{q \in \mathcal{Q}} \; \mathcal{D}_{\mathrm{KL}}(q(\mathbf{Y}) || \; p_\theta(\mathbf{Y}|\mathbf{X}))$$

***Valid* posteriors:** $\mathcal{Q} = \{q(\mathbf{Y}) : \mathbf{E}_q[\phi] \approx \mathbf{b}\}$

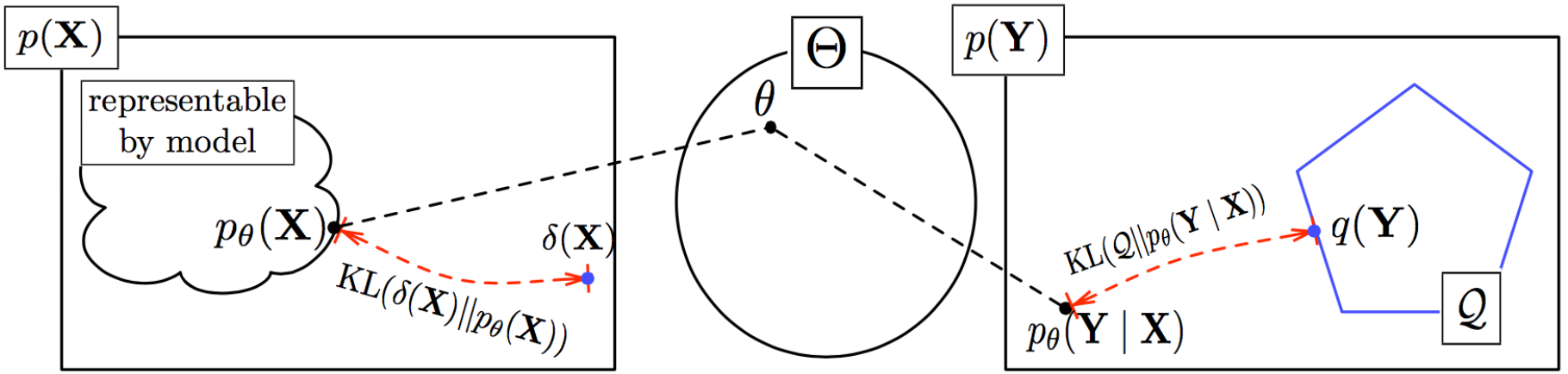
[e.g: *q(Y)* that assign 25% articles to "politics"]

Posterior Regularization

J. Graça, K. Ganchev, B. Taskar (2007).

University of Pennsylvania (2007)

Idea: Regularize data likelihood with *valid* posteriors



Negative Log-Likelihood

Posterior Regularization

Optimization: Expectation Maximization

Objective:optimize marginal likelihood $\mathcal{L}(\theta) = \log \sum_{\mathbf{Y}} p_\theta(\mathbf{X}, \mathbf{Y})$

By Jensen's inequality, we define a lower-bound $F(q,\theta)$ as

$$\mathcal{L}(\theta) = \log \sum_{\mathbf{Y}} q(\mathbf{Y}) \frac{p_\theta(\mathbf{X}, \mathbf{Y})}{q(\mathbf{Y})} \geq \sum_{\mathbf{Y}} q(\mathbf{Y}) \log \frac{p_\theta(\mathbf{X}, \mathbf{Y})}{q(\mathbf{Y})} = F(q, \theta).$$

We can re-write $F(q,\theta)$ as

$$\begin{aligned} F(q, \theta) &= \sum_{\mathbf{Y}} q(\mathbf{Y}) \log(p_\theta(\mathbf{X}) p_\theta(\mathbf{Y}|\mathbf{X})) - \sum_{\mathbf{Y}} q(\mathbf{Y}) \log q(\mathbf{Y}) \\ &= \mathcal{L}(\theta) - \sum_{\mathbf{Y}} q(\mathbf{Y}) \log \frac{q(\mathbf{Y})}{p_\theta(\mathbf{Y}|\mathbf{X})} \\ &= \mathcal{L}(\theta) - \mathbf{KL}(q(\mathbf{Y}) || p_\theta(\mathbf{Y}|\mathbf{X})). \end{aligned}$$

Optimization: Expectation Maximization

Objective:optimize marginal likelihood

$$\mathcal{L}(\theta) = \log \sum_{\mathbf{Y}} p_\theta(\mathbf{X}, \mathbf{Y}) \qquad F(q, \theta) = \mathcal{L}(\theta) - \mathbf{KL}(q(\mathbf{Y}) || p_\theta(\mathbf{Y}|\mathbf{X}))$$

Expectation Maximization:

$$\mathbf{E}: q^{t+1} = \arg\max_q F(q, \theta^t) = \arg\min_q \mathbf{KL}(q(\mathbf{Y}) \parallel p_{\theta^t}(\mathbf{Y} \mid \mathbf{X})),$$

$$\mathbf{M}: \theta^{t+1} = \arg\max_\theta F(q^{t+1}, \theta) = \arg\max_\theta \mathbf{E}_{q^{t+1}} [\log p_\theta(\mathbf{X}, \mathbf{Y})].$$

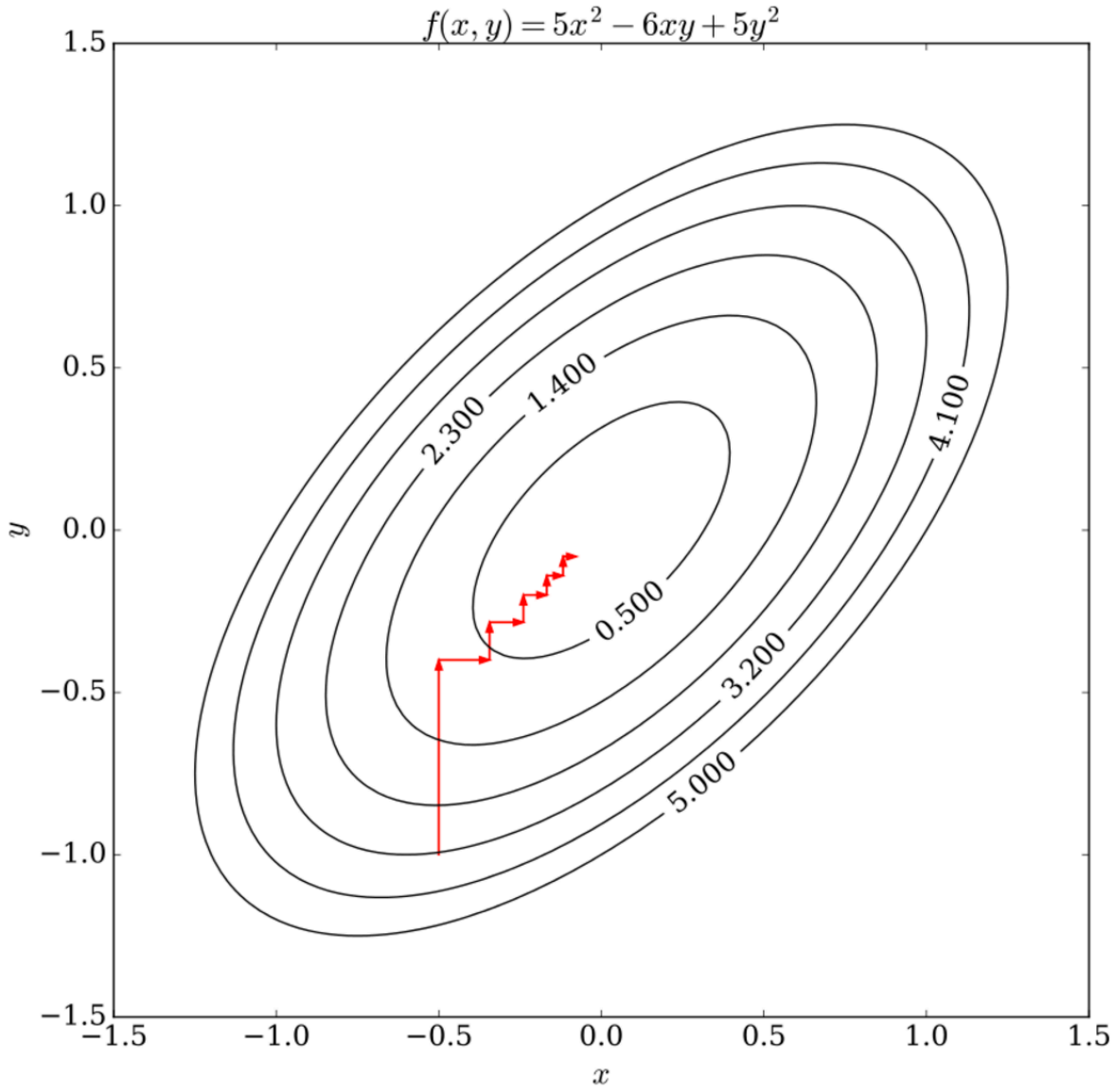
We can easily see:

$$F(q^{t+2}, \theta^{t+1}) \geq F(q^{t+1}, \theta^{t+1}) \geq F(q^{t+1}, \theta^t)$$

A More Illustrative Example

Minimize:

$f(x,y)=5x^2-6x*y+5y^2$



Optimizing Posterior Regularization

Idea : EM algorithm with **valid posteriors**

PR Objective: $J_{\mathcal{Q}}(\theta) = \max_{q \in \mathcal{Q}} F(q, \theta) = \mathcal{L}(\theta) - \min_{q(\mathbf{Y}) \in \mathcal{Q}} \mathbf{KL}(q(\mathbf{Y}) || p_\theta(\mathbf{Y}|\mathbf{X}))$

Constrained EM: constrain with **valid posteriors**

$$\mathcal{Q} = \{q(\mathbf{Y}) : \mathbf{E}_q[\phi] \approx \mathbf{b}\}$$

$$\mathbf{E}' : q^{t+1} = \arg\max_{q \in \mathcal{Q}} F(q, \theta^t) = \arg\min_{q \in \mathcal{Q}} \mathbf{KL}(q(\mathbf{Y}) \parallel p_{\theta^t}(\mathbf{Y}|\mathbf{X}))$$

$$\mathbf{M} : \theta^{t+1} = \arg\max_{\theta} F(q^{t+1}, \theta) = \arg\max_{\theta} \mathbf{E}_{q^{t+1}}[\log p_\theta(\mathbf{X}, \mathbf{Y})].$$

How to Solve the Min Sub-problem?

The Minimization Problem:

$$\min_{q,\xi} \quad \mathbf{KL}(q(\mathbf{Y}) \parallel p_\theta(\mathbf{Y}|\mathbf{X})) \qquad \text{s.t.} \quad \mathbf{E}_q[\phi(\mathbf{X},\mathbf{Y})] - \mathbf{b} \leq \xi; \quad ||\xi||_\beta \leq \varepsilon.$$

The Dual Problem:

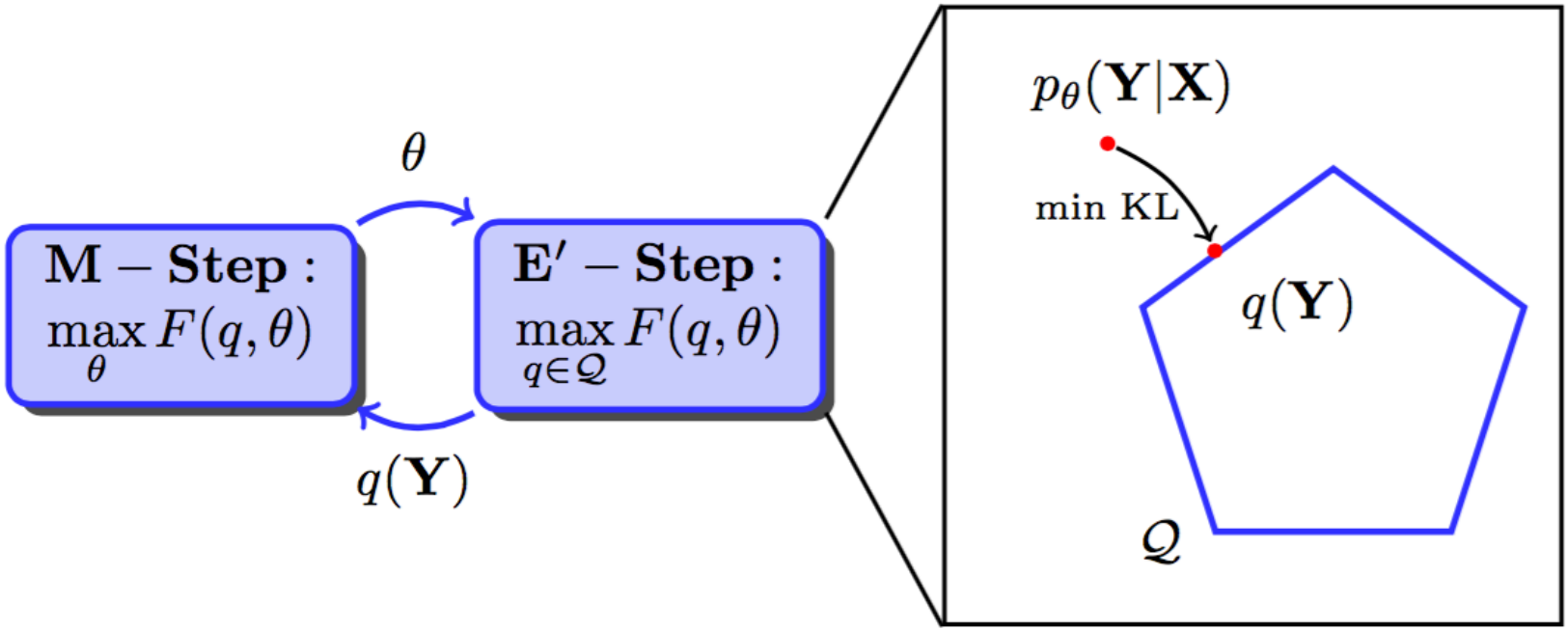
$$\max_{\lambda \geq 0} \quad -\mathbf{b}\cdot\lambda - \log Z(\lambda) \; - \varepsilon\,||\lambda||_{\beta^*}.$$

The Solution:

$$q^*(\mathbf{Y}) = \frac{p_\theta(\mathbf{Y}|\mathbf{X})\exp\{-\lambda^*\cdot\phi(\mathbf{X},\mathbf{Y})\}}{Z(\lambda^*)}$$

$$Z(\lambda^*) = \Sigma_{\mathbf{Y}}\, p_\theta(\mathbf{Y}|\mathbf{X})\exp\{-\lambda^*\cdot\phi(\mathbf{X},\mathbf{Y})\}$$

Optimizing Posterior Regularization



$$\mathbf{M}: \theta^{t+1} = \arg\max_{\theta} F(q^{t+1}, \theta) = \arg\max_{\theta} \mathbf{E}_{q^{t+1}} [\log p_\theta(\mathbf{X}, \mathbf{Y})]$$

$$\mathbf{E}': q^{t+1} = \arg\max_{q \in \mathcal{Q}} F(q, \theta^t) = \arg\min_{q \in \mathcal{Q}} \mathbf{KL}(q(\mathbf{Y}) \,\|\, p_{\theta^t}(\mathbf{Y}|\mathbf{X}))$$

Posterior Regularization

Hard constraints:

$$\max_{\theta} \quad \mathcal{L}(\theta) \; - \; \min_{q \in \mathcal{Q}} \; \mathcal{D}_{\mathrm{KL}}(q(\mathbf{Y}) || \; p_\theta(\mathbf{Y}|\mathbf{X}))$$

$$\mathcal{Q} = \left\{ q(\mathbf{Y}) : \|\mathbf{E}_q[\phi(\mathbf{Y})] - \mathbf{b}\|_2^2 \leq \epsilon \right\}$$

Soft constraints:

$$\max_{\theta} \quad \mathcal{L}(\theta) \; - \; \min_{q} \left(\begin{array}{r} \mathcal{D}_{\mathrm{KL}}(q(\mathbf{Y}) || \; p_\theta(\mathbf{Y}|\mathbf{X})) + \\ \alpha \, \|\mathbf{E}_q[\phi(\mathbf{Y})] - \mathbf{b}\|_2^2 \end{array} \right)$$

Applications with Posterior Regularization

Sentence-level Sentiment Classification

Yang and Cardie,ACL 2014.

Idea: incur context-aware posterior constraints for classification

Problem: given a sequence of sentences X, predict the labels Y

Model: Conditional Random Fields

$$p_\theta(\mathbf{y}|\mathbf{x}) = \frac{\exp(\theta \cdot f(\mathbf{x}, \mathbf{y}))}{Z_\theta(\mathbf{x})}$$

Data likelihood:

$$\max_\theta \mathcal{L}(\theta) = \max_\theta \sum_{(\mathbf{x},\mathbf{y})} \log p_\theta(\mathbf{y}|\mathbf{x}) - \frac{||\theta||_2^2}{2\delta^2}$$

Objective:

$$\max_\theta \mathcal{L}(\theta) - \min_{q \in \mathcal{Q}} \{KL(q(\mathbf{Y})||p_\theta(\mathbf{Y}|\mathbf{X}))$$

Soft constraints:

$$+ \beta||E_q[\phi(\mathbf{X}, \mathbf{Y})] - \mathbf{b}||_2^2\}$$

$$\mathcal{Q} = \{q(\mathbf{Y}) : E_q[\phi(\mathbf{X}, \mathbf{Y})] = \mathbf{b}\}$$

Sentence-level Sentiment Classification

Yang and Cardie,ACL 2014.

Types	Description and Examples
Lexical patterns	The sentence containing a polar lexical pattern w tends to have the polarity indicated by w. Example lexical patterns are *annoying, hate, amazing, not disappointed, no concerns, favorite, recommend*.
Discourse Connectives (clause)	The sentence containing a discourse connective c which connects its two clauses that have opposite polarities indicated by the lexical patterns tends to have neutral sentiment. Example connectives are *while, although, though, but*.
Discourse Connectives (sentence)	Two adjacent sentences which are connected by a discourse connective c tends to have the same polarity if c indicates a *Expansion* or *Contingency* relation, e.g. *also, for example, in fact, because* ; opposite polarities if c indicates a *Comparison* relation, e.g. *otherwise, nevertheless, however*.
Coreference	The sentences which contain coreferential entities appeared as targets of opinion expressions tend to have the same polarity.
Listing patterns	A series of sentences connected via a listing tend to have the same polarity.
Global labels	The sentence-level polarity tends to be consistent with the document-level polarity.

Lexical constraints: A sentence label has its expected value of its sentiment words

$$\phi_w(x, y) = \sum_i f_w(x_i, y_i)$$

Discourse constraints: Contradictory sentiment transition should be small

$$\phi_{c,s}(x, y) = \sum_i f_{c,s}(x_i, y_i, y_{i-1})$$

Aspect Phrase Clustering

Zhao, Huang, et al. EMNLP 2014.

Li Zhao, Minlie Huang, Haiqiang Chen, Junjun Cheng, Xiaoyan Zhu. Clustering Aspect-related Phrases by Leveraging Sentiment Distribution Consistency. EMNLP 2014, October 25–29, 2014 — Doha, Qatar.

Aspect Phrase Clustering

Zhao, Huang, et al. EMNLP 2014.

Goal：clustering aspect phrases that refer to the same product property

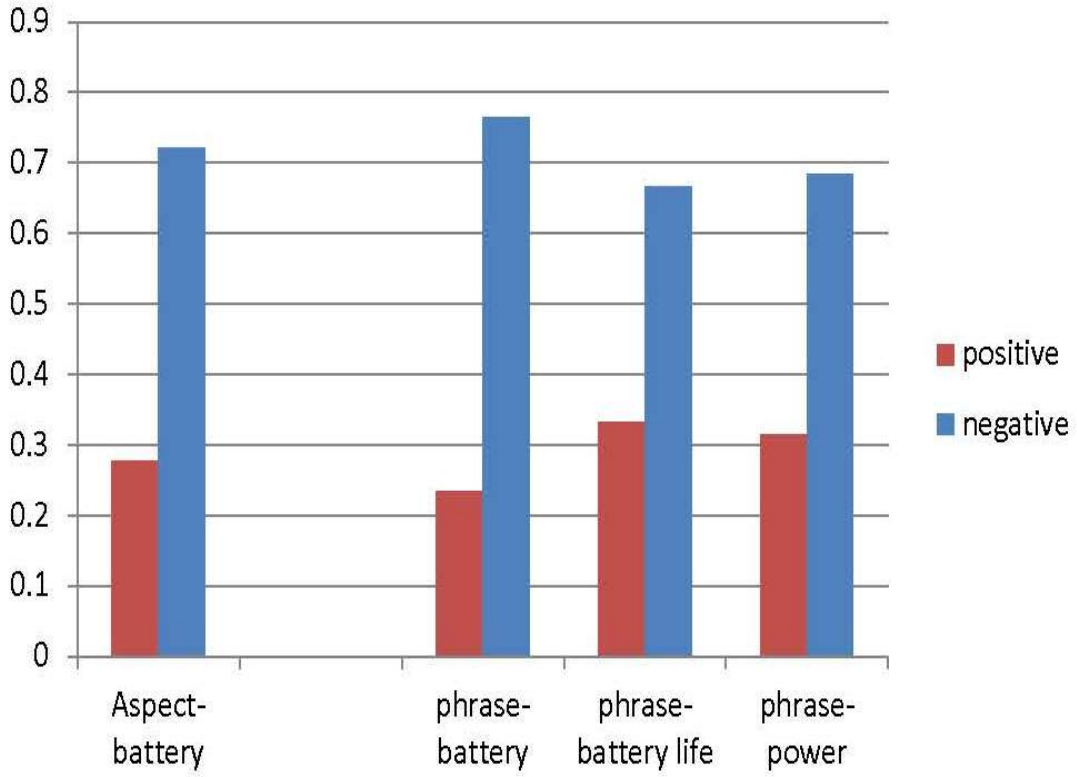
- aspect “battery”: {“battery”, “battery life”, “power”…}

Semi-structured Reviews:

Pros:	***LCD***, *nice* ***touch screen***, *longer* **battery life**
Cons:	*Horrible* ***picture quality***
Review:	The ***touch screen*** was the selling feature for me. The ***LCD touch screen*** is nice and large. This camera also has very impressive ***battery life***. However the ***picture quality*** is very grainy.

The same aspect can not be in Pros and Cons at the same time!

Sentiment Distribution Consistency

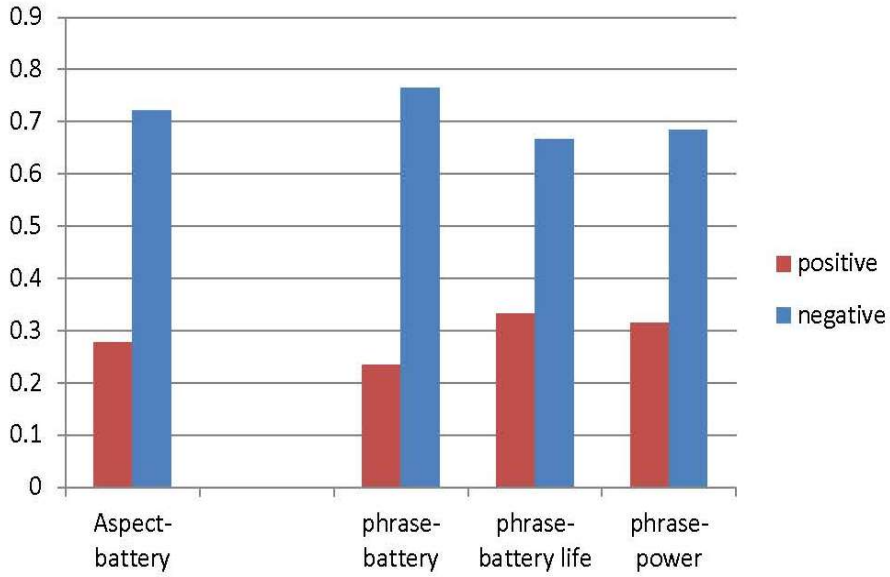


	Pros	Cons
phrase-battery	19	62
phrase-battery life	37	74
phrase-power	17	37
…	…	…

The sentiment distribution of aspect "battery" and its related-phrases on ***a product k*** with a large amount of reviews.

Sentiment Distribution Consistency

Different phrases of the same aspect tend to have the same sentiment distribution, or to have statistically close estimated distributions.



	Pros	Cons
phrase-battery	19	62
phrase-battery life	37	74
phrase-power	17	37
…	…	…

The sentiment distribution of aspect “battery” and its related-phrases

On ***a product k*** with a large amount of reviews.

Some Statistics……

X_{ai} : a random variable indicating the sentiment on aspect a_i

- 1: aspect a_i receives positive comments; 0 negative

$$X_{a_i} \sim Bernoulli(p_{a_i})$$

- p_{ai}: the prob.of a_i being rated positively

Sentiment Distribution Consistency

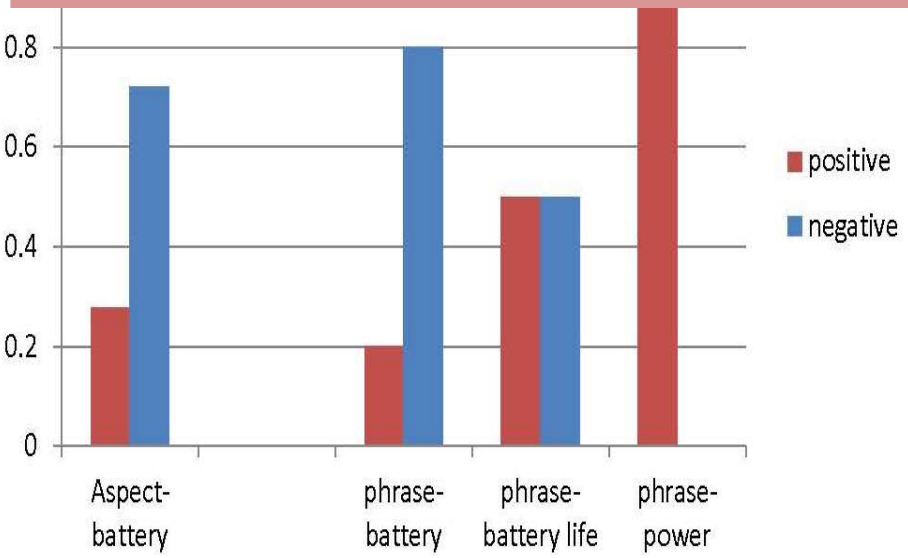
For an aspect phrase f_j, if $f_j \in a_i$

then: $X_{f_j} \sim Bernoulli(p_{a_i})$

However…

When the number of reviews is limited…

The estimated sentiment distribution is unreliable!



		Cons
battery	1	4
phrase-battery life	2	2
phrase-power	1	0
…	…	…

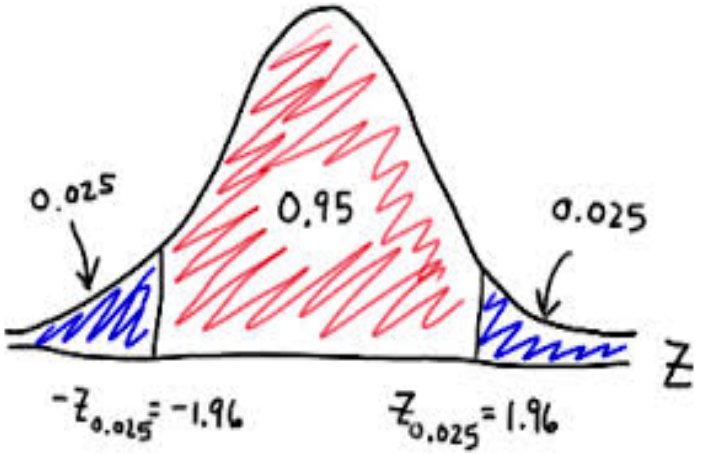
The sentiment distribution of aspect "battery" and its related-phrases on ***a product k'*** with a small mumber of reviews.

Confidence Interval

Idea：use interval estimation to generate flexible constraints

	pros	cons
Aspect phrase1	21	10

Phrase distribution P_{fj}=<0.67, 0.33>



Aspect Distribution

$$\phi = z_{ji}|u_{ik} - \hat{s}_{jk}| \leq d_{jk}, \forall i, j, k$$

$$z_{ji} = \begin{cases} 1 & ;\text{if } f_j \in a_i \\ 0 & ;\text{otherwise} \end{cases}$$

$$d_{jk} = C\frac{\hat{\sigma}_{jk}}{\sqrt{n_{jk}}}.$$

MLE sentiment rating of aspect a_i on product k

MLE sentiment rating of phrase f_j on product k

Aspect Phrase Clustering

Data likelihood: $$p_\theta(D) = \prod_{j=1}^{|D|} p_\theta(d_j) = \prod_{j=1}^{|D|} \sum_{y_j \in A} p_\theta(d_j, y_j)$$

Data likelihood with MNB(each phrase is represented by a context document):

$$p_\theta(d_j, y_j = a_i) = p(a_i) \prod_{k=1}^{|d_j|} p(w_{d_j,k}|a_i)$$

Valid posterior constraints:

$$Q = \{q(Y) : q(y_j = a_i)|u_{ik} - \hat{s}_{jk}| \leq d_{jk}, \forall i, j, k\}$$

Aspect distribution: $$u_{ik} = \frac{1}{\sum_{j=1}^{|D|} n_{jk} p_\theta(a_i|d_j)} \sum_{j=1}^{|D|} n_{jk} p_\theta(a_i|d_j) \hat{s}_{jk}$$

Prob. of a phrase belonging to an aspect : $$p_\theta(a_i|d_j) = \frac{p(a_i) \prod_{k=1}^{|d_j|} p(w_{d_j,k}|a_i)}{\sum_{r=1}^{|A|} p(a_r) \prod_{k=1}^{|d_j|} p(w_{d_j,k}|a_r)}$$

Aspect Phrase Clustering - Experiments

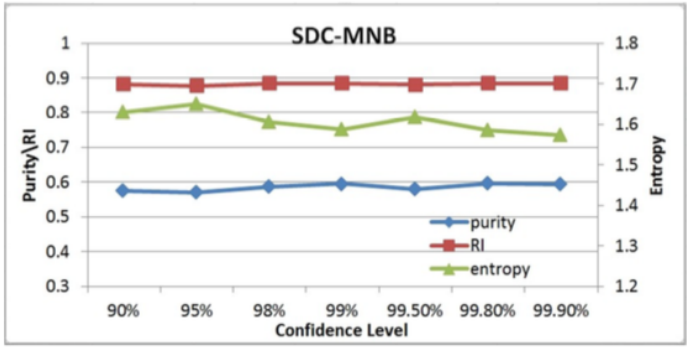
Purity, Rand Index, Entropy

	Camera			Cellphone			Laptop			MP3		
	P	RI	E	P	RI	E	P	RI	E	P	RI	E
Kmeans	43.48%	83.52%	2.098	48.91%	84.80%	1.792	43.46%	87.11%	2.211	40.00%	70.98%	2.047
L-EM	54.89%	87.07%	**1.690**	51.96%	86.64%	1.456	48.94%	84.53%	2.039	44.24%	75.37%	1.990
LDA	36.84%	83.28%	2.426	48.65%	85.33%	1.833	35.02%	83.53%	2.660	36.12%	76.08%	2.296
Constraint-LDA	43.30%	86.01%	2.216	47.89%	86.04%	1.974	32.35%	84.86%	2.676	50.70%	81.42%	1.924
SDC-MNB	**56.42%**	**88.16%**	1.725	**67.95%**	**90.62%**	**1.266**	**55.52%**	**90.72%**	**1.780**	**58.06%**	**83.57%**	**1.578**

Compared to supervised models

	Purity	RI	Entropy
MNB-5%	53.21%	85.77%	1.854
MNB-10%	**59.55%**	**86.70%**	**1.656**
MNB-15%	66.06%	88.39%	1.449
L-Kmeans-10%	53.54%	86.15%	1.745
L-Kmeans-15%	**57.00%**	**86.89%**	**1.643**
L-Kmeans-20%	60.97%	87.63%	1.528
SDC-MNB	**59.49%**	**88.26%**	**1.580**

Sensitivity analysis



Constraint Effect

	$\#(d_{jk} < 0.2)$	$\#(0.2 < d_{jk} < 1)$	purity gain
Camera	3.02	8.78	1.53%
Cellphone	17.29	30.5	15.99%
Laptop	4.6	13.22	6.58%
MP3MP4	6.1	10.7	13.82%

Sentiment Extraction

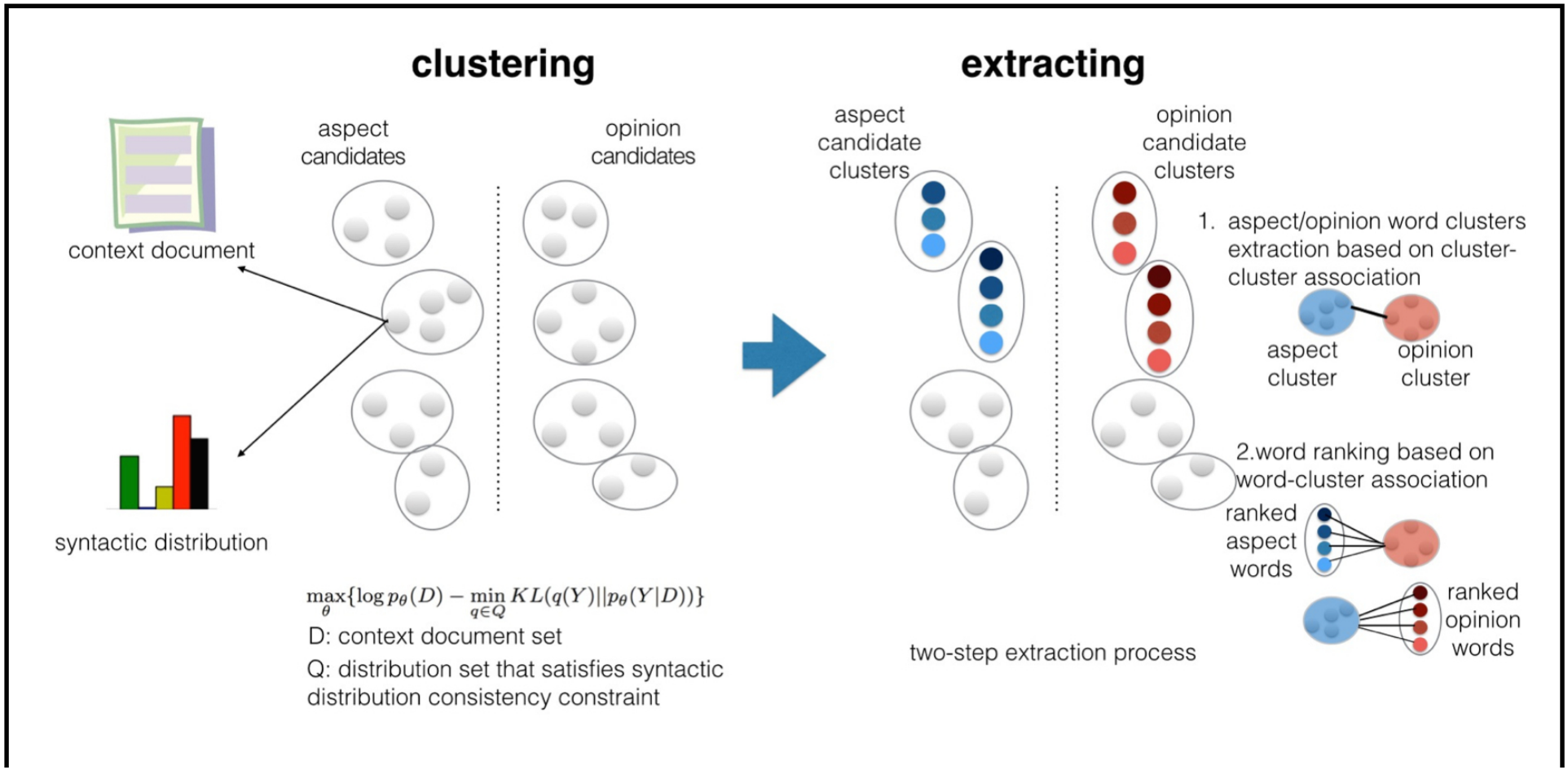
Zhao, Huang, et al. CIKM 2015.

Li zhao, Minlie Huang, Xiaoyan Zhu. Sentiment Extraction by Leveraging Aspect-Opinion Association Structure. CIKM 2015, Oct 19-23, Melbourne, Australia.

Sentiment Extraction

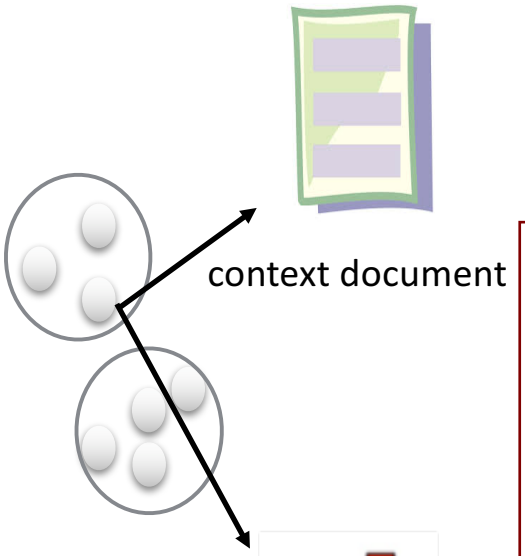
Zhao, Huang, et al. CIKM 2015.

What if we do not have semi-structured data?

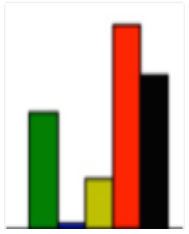


Clustering with Syntactic Constraints

Zhao, Huang, et al. CIKM 2015.

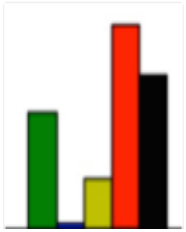


context document



syntactic distribution

$$Syn(ac = screen)$$
$$= (Count(\overleftarrow{nsubj}\ clear), Count(\overleftarrow{nsubj}\ fine),$$
$$..., Count(\overrightarrow{rcmod}\ expensive), ...)^T$$
$$= (73, 31, ..., 0, ...)^T$$



syntactic distribution

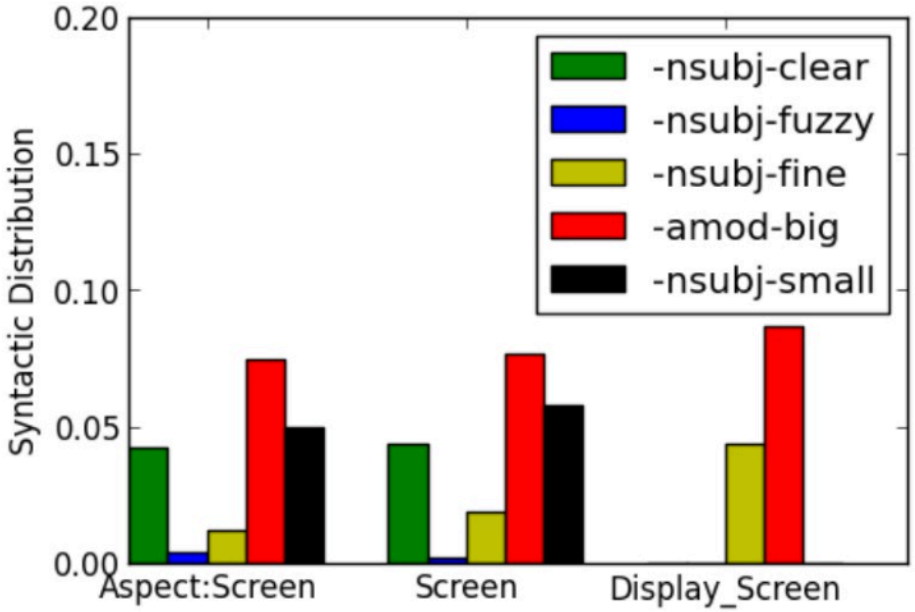
Aspect cluster distribution

$$Syn(acc_i) \sim Multinomial(n_i, \vec{p}_i)$$

Aspect cluster

Aspect cluster distribution

Clustering with Syntactic Constraints



	Count
Aspect:Screen	3,267
Screen	1,670
Display-Screen	23

Pearson Chi-square Test:

Expected Observed

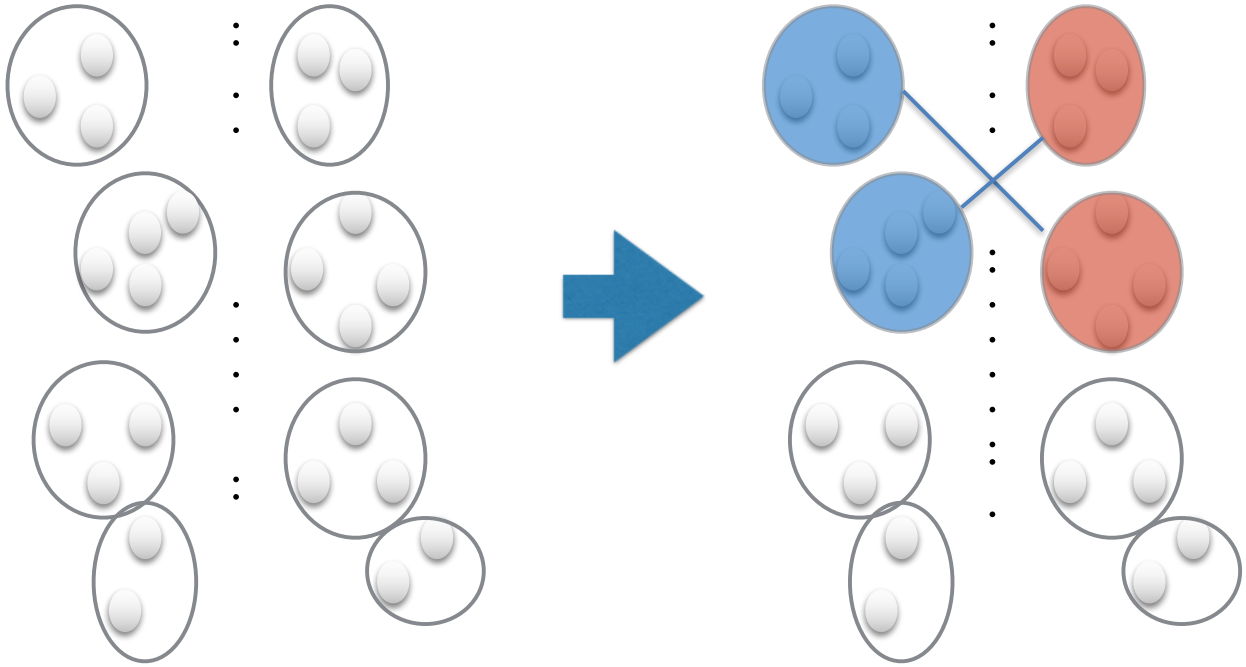
$$\chi^2(acc_i, ac_j) = \sum_{k=1}^{dim(\vec{p}_i)} \frac{(n_j\vec{p}_i^{(k)} - Syn(ac_j)^{(k)})^2}{n_j\vec{p}_i^{(k)}} \leq \chi^2_{dim(\vec{p}_i)-1,\alpha}$$

$$z_{ij}\chi^2(acc_i, ac_j) \leq \chi^2_{dim(\vec{p}_i)-1,\alpha} \qquad z_{ij} = \begin{cases} 1 & ; \text{if } ac_j \in acc_i \\ 0 & ; \text{otherwise} \end{cases}$$

Valid posterior constraints:

$$\max_\theta \{\log p_\theta(D) - \min_{q \in Q} KL(q(Y) \| p_\theta(Y|D))\} \qquad Q = \{q(Y) : E_q[z_{ij}\chi^2(acc_i, ac_j)] \leq \chi^2_{dim(\vec{p}_i)-1,\alpha}, \forall i, j\}$$

Extraction from Clusters



Likelihood Ratio Test

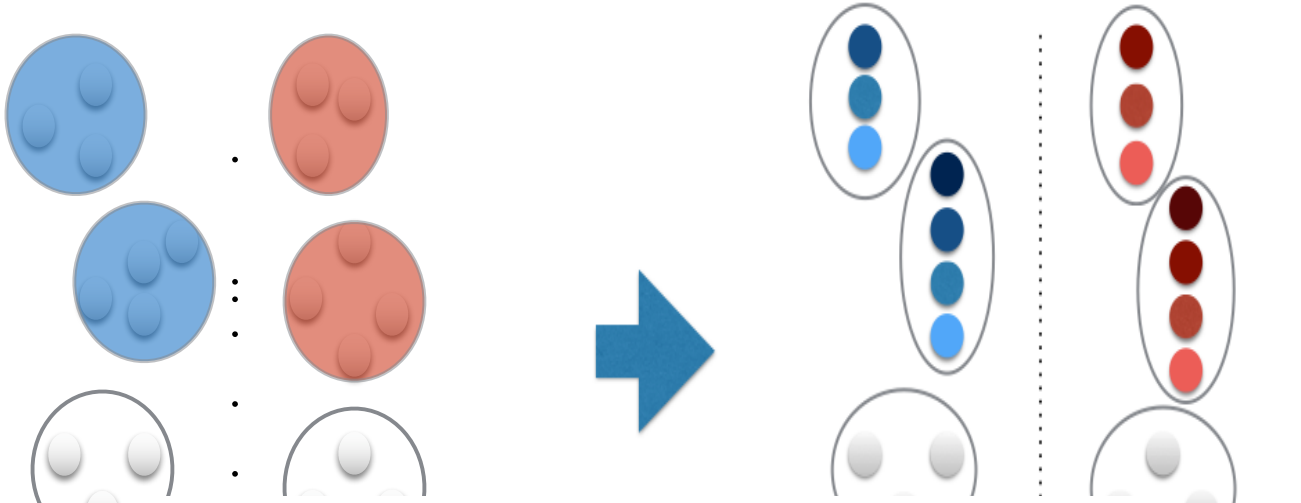
Corpus statistics	AC_i	$\overline{AC_i}$
OC_j	$k_1(AC_i, OC_j)$	$k_2(AC_i, \overline{OC_j})$
$\overline{OC_j}$	$k_3(\overline{AC_i}, OC_j)$	$k_4(\overline{AC_i}, \overline{OC_j})$

$$
\begin{aligned}
LRT(AC_i, OC_j) &= f(k_1, k_2, k_3, k_4) \\
&= 2[logL(p1, k1, n1) + logL(p2, k2, n2) \\
&\quad - logL(p, k1, n1) - logL(p, k2, n2)]
\end{aligned}
$$

$$
\begin{aligned}
&L(p, k, n) = p^k(1-p)^{n-k}; n_1 = k_1 + k_3; \\
&n_2 = k_2 + k_4; p_1 = k_1/n_1; p_2 = k_2/n_2; \\
&p = (k_1 + k_2)/(n_1 + n_2;)
\end{aligned}
$$

Extraction from Clusters

Rank words within a cluster based on the same measure



语料集上的统计信息	C_i	$\overline{C_i}$
T_j	$k_1(C_i, T_j)$	$k_2(C_i, \overline{T_j})$
$\overline{T_j}$	$k_3(\overline{C_i}, T_j)$	$k_4(\overline{C_i}, \overline{T_j})$

Sentiment Extraction - Experiments

Dataset

	Camera	Cellphone	Laptop	MP3/MP4
#Products	449	694	702	329
#Reviews	101,235	579,402	102,439	129,471
#Aspect Candidates	3,686	6,360	4,373	3,724
#Aspect Words	894	1,321	993	930
#Aspect	27	26	25	27
#Opinion Candidates	1,614	2,235	1,657	1,617
#Opinion Words	806	880	791	792
#Opinion	28	28	28	28

Good topic: For each topic (or cluster in our setting), we judge it as a "good topic" if the top 15 words contain at least 5 synonymous aspect words.

	Camera	Cellphone	Laptop	MP3/MP4
LDA	3.8/27	4.0/26	6.8/25	4.8/27
AMC	8.6/27	11.4/26	13.8/25	11.8/27
SAS	6.0/27	6.0/26	15.0/25	9.0/27
Ours	**16.8**/27	**15.6**/26	**15.4**/25	**18.0**/27

Table 4: Comparison with topic model baselines : each cell denotes the number of good topics/ all extracted topics.

Our approach can extract more good topics with better precision.

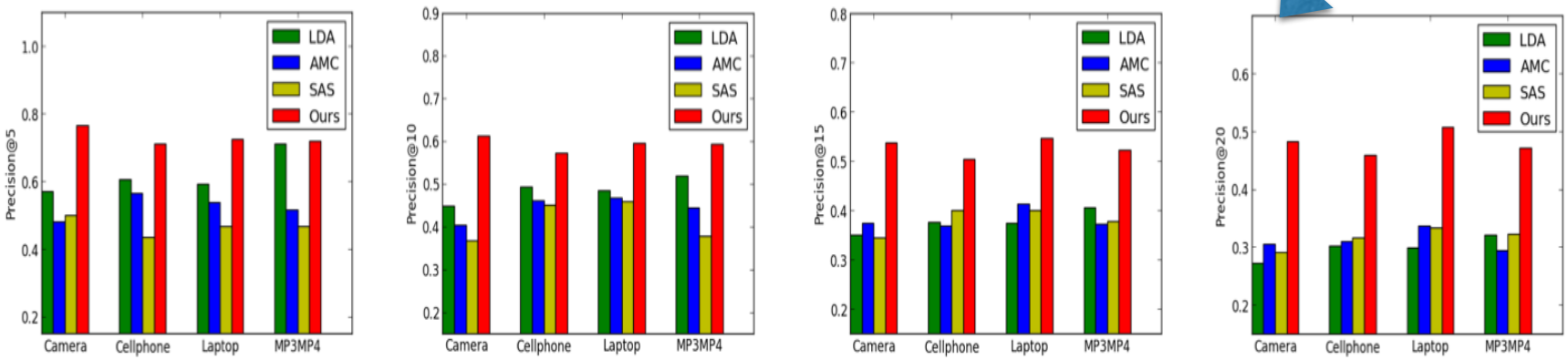


Figure 5: Comparison with topic model baselines : p@k of good topics across different domains. k = 5,10,15,20.

Sentiment Classification in Semi-Supervised Settings

Zhao, Huang, et al. AAAI 2016.

Li Zhao, Minlie Huang, Ziyu Yao, Xiaoyan Zhu. Semi-supervised Multinomial Naive Bayes for Text Classification by Leveraging Word-level Statistical Constraint, AAAI 2016

Sentiment Classification in Semi-Supervised Settings

Zhao, Huang, et al. AAAI 2016.

Motivation: Semi-supervised learning may not be stable.

G. S. Mann and A. McCallum. Simple, robust, scalable semi-supervised learning via expectation regularization. In *Proc. ICML*, 2007.

speech tagging (Klein & Manning, 2004). However, while EM sometimes works very well, it can be fragile, finding solutions that are worse than the equivalent supervised model. Cozman and Cohen (2006) discuss

Dataset	EM	MNB
New3	70.64±2.00 •	72.69±2.52 •
Ohscal	86.76±1.96	87.32±1.64
20news	86.95±3.42	85.03±1.86 •
Sraa	84.16±5.87	83.09±1.78 •
Economics	71.86±0.99 •	74.67±1.77 •
Market	91.00±1.38 •	90.92±2.40 •
Government	62.38±1.96 •	64.82±1.87 •
Corporate	55.24±2.71 •	60.86±1.72 •
Average	76.13	77.42

Jiang Su, Jelber Sayyad Shirab, and Stan Matwin. 2011. Large scale text classification using semisupervised multinomial naive bayes. In Lise Getoor and Tobias Scheffer, editors, *ICML*, pages 97–104. Omnipress.

MNB with EM

Multinomial Naïve Bayes

$$P(c|d) = \frac{P(c)\prod_{i=1}^{|V|} P(w_i|c)^{N_{d,w_i}}}{P(d)}$$

Multinomial Naïve Bayes with unlabeled data

$$\max_{\theta} \sum_{d \in L} logP(c,d) + \sum_{d \in U} logP(d)$$

Word Class Distribution with MNB

Word Class Distribution

For each word w, how likely does it appear in different class?

MLE on labeled data $(p_w^c)_l = \frac{N_w^c}{\sum_i N_w^i}$

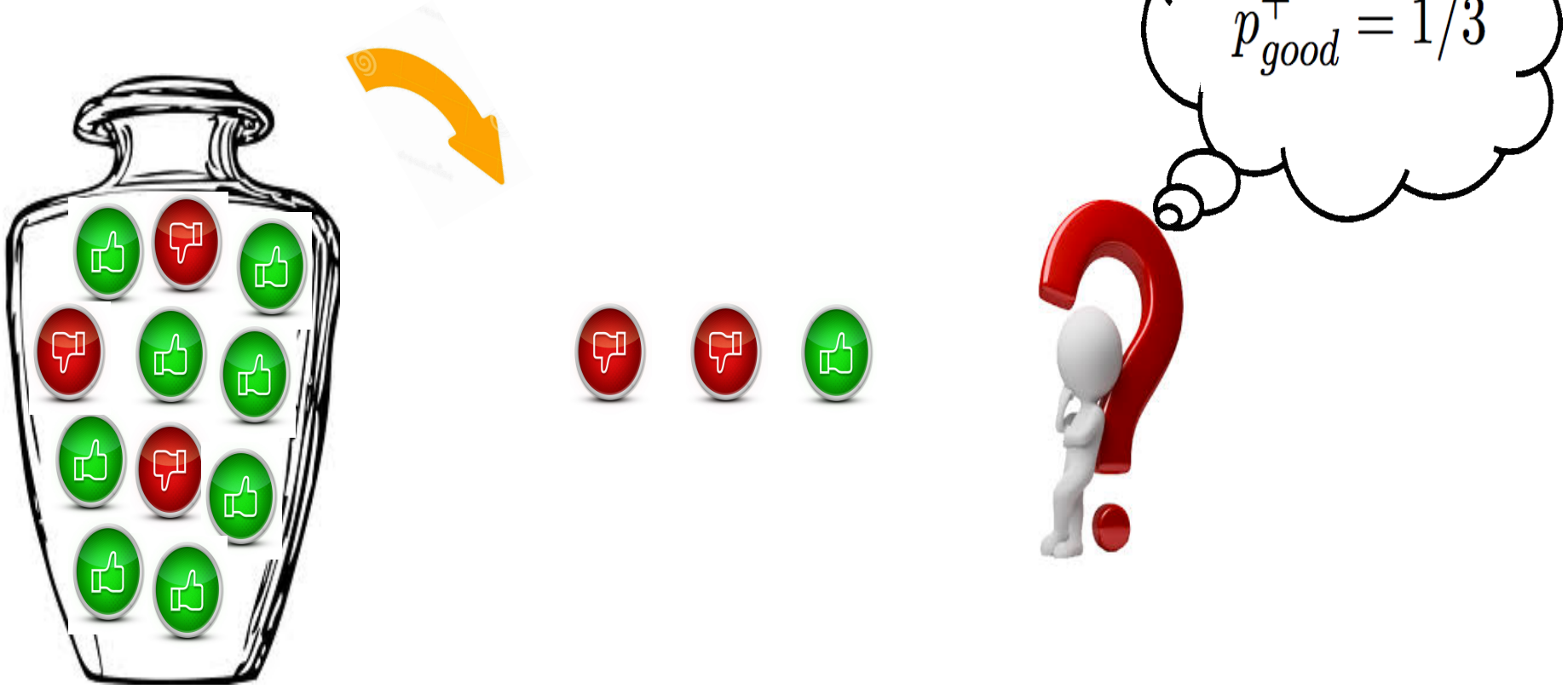
	N^+_{loves}	N^-_{loves}	p^+_{loves}
labeled data	18	2	0.9

Maximum Likelihood Estimation(MLE)

$$(p_w^c)_l = \frac{N_w^c}{\sum_i N_w^i}$$

jar for “good”

three samples

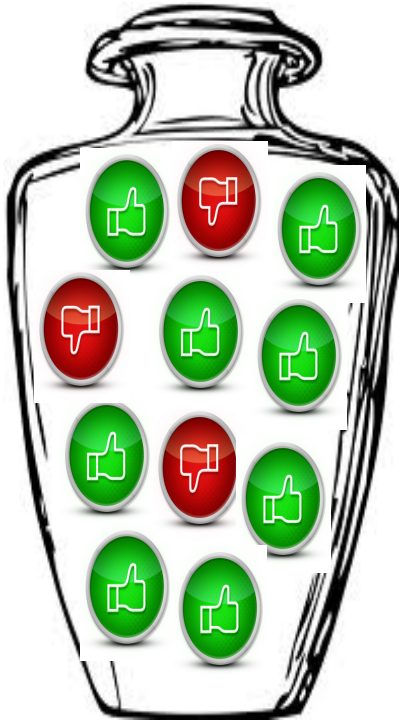


Maximum Likelihood Estimation(MLE)

$$(p_w^c)_l = \frac{N_w^c}{\sum_i N_w^i}$$

jar for “good”

three samples



$p^+_{good} = 1/3$

Estimated parameters for less frequent word are unreliable!

MNB-EM

MLE on labeled data and EM's estimation P(c|d) for unlabeled data

$$(p_w^c)_u = \frac{\sum_d N_{d,w} \times P(c|d)}{\sum_d N_{d,w}}$$

$(p_w^c)_u$ may be different from $(p_w^c)_l$.

But how much could $(p_w^c)_u$ be different from $(p_w^c)_l$?

Interval Estimation

Wilson interval $$CI = \frac{(p_w^c)_l + \frac{z^2_{\alpha/2}}{2N_w} \pm z_{\alpha/2}\sqrt{[(p_w^c)_l(1-(p_w^c)_l) + z^2_{\alpha/2}/4N_w]/N_w}}{(1+z^2_{\alpha/2}/N_w)}$$

Generate lower and upper bound $$lower(p_w^c) \leq (p_w^c)_u \leq upper(p_w^c)$$

on labeled data

	N_w^+	N_w^-	置信区间(p_w^+)
"loves"	18	2	[0.6990,0.9721]
"worthless"	2	1	[0.2076,0.9385]

Frequent w has relatively reliable estimation.
For frequent w, P_u should not be too different from P_l.

Document Posterior Constraint

Word class distribution should be maintained on both supervised and unsupervised data

$$(p_w^c)_u = \frac{\sum_d N_{d,w} \times P(c|d)}{\sum_d N_{d,w}}$$

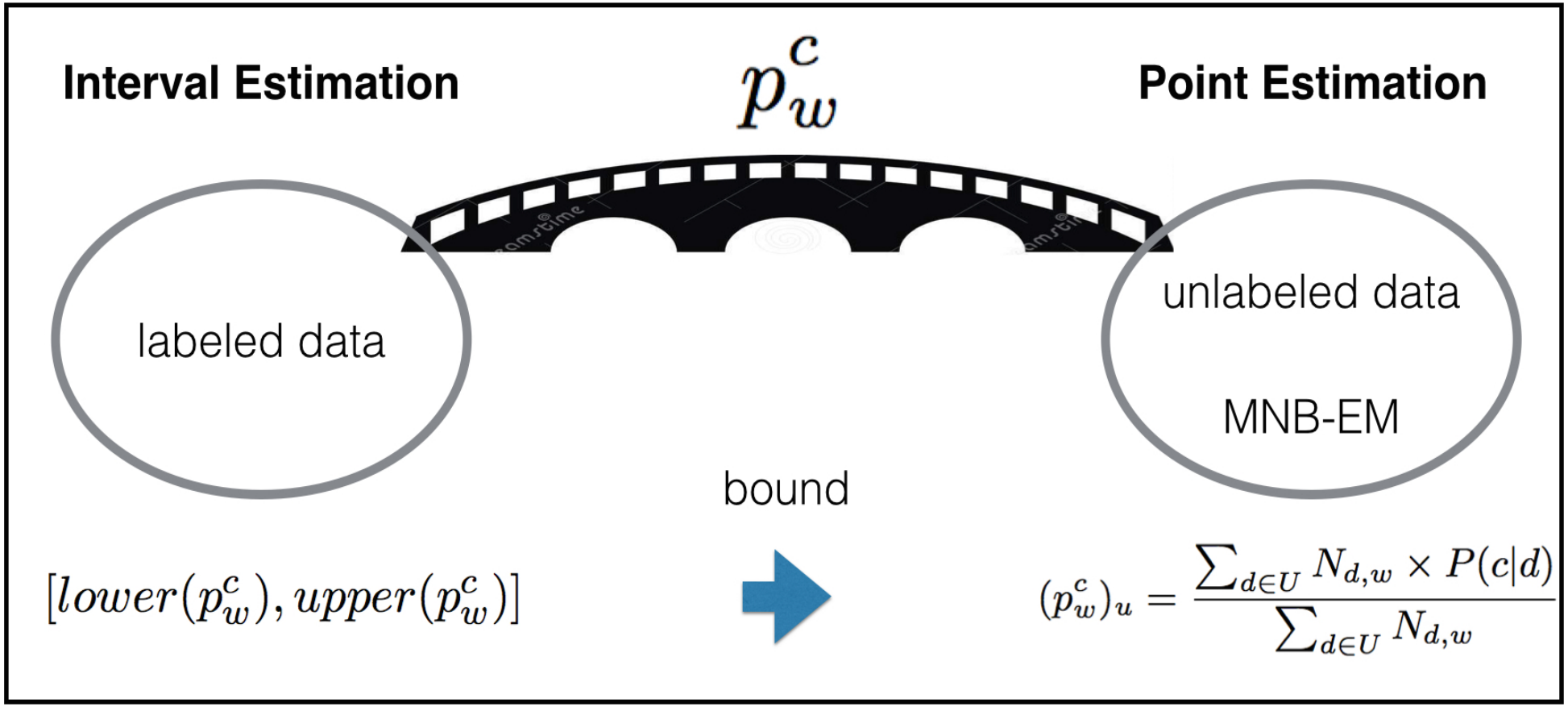
$$lower(p_w^c) \leq (p_w^c)_u \leq upper(p_w^c)$$

$$lower(p_w^c) \times N_w \leq \sum_d N_{d,w} \times P(c|d) \leq upper(p_w^c) \times N_w$$

Document Posterior Constraint

Word class distribution should be maintained

on both labeled and unlabeled data



Experiments

Dataset

Dataset	#Class	#Instance	Positive(%)	$\|V\|$
Ohscal	10	11,162	-	11,466
Reuters	8	7,674	-	17,387
WebKB	4	4,199	-	7,770
20News	20	18,828	-	24,122
Kitchen	2	19,856	79.25	10,442
Electronics	2	23,009	78.06	12,299
Toys&Games	2	13,147	80.46	8,448
Dvd	2	124,438	85.87	56,713

Comparison of Macro-F1 for Topic/Sentiment Classification

number of labeled documents $T_l = 64$

dataset	MNB	MNB-EM	SFE	MNB-WSC
Ohscal	0.4644•	0.5353	0.4591•	0.5417
Reuters	0.4824•	0.4905	0.5330	0.5359
WebKB	0.6589•	0.6674	0.6911	0.6945
20News	0.3177•	0.4242	0.3204•	0.4429

number of labeled documents $T_l = 128$

dataset	MNB	MNB-EM	SFE	MNB-WSC
Ohscal	0.5456•	0.5878	0.5435•	0.6022
Reuters	0.5843•	0.5938	0.6102	0.6291
WebKB	0.7157•	0.7250	0.7332	0.7439
20News	0.4115•	0.5322	0.4302•	0.5327

number of labeled documents $T_l = 256$

dataset	MNB	MNB-EM	SFE	MNB-WSC
Ohscal	0.5955•	0.6166•	0.5956•	0.6418
Reuters	0.6789	0.6826	0.6862	0.7054
WebKB	0.7526	0.7425	0.7637	0.7677
20News	0.5161•	0.6175	0.5443•	0.6215

number of labeled documents $T_l = 512$

dataset	MNB	MNB-EM	SFE	MNB-WSC
Ohscal	0.6420•	0.6404•	0.6399•	0.6736
Reuters	0.7647	0.7714	0.7656	0.7723
WebKB	0.7768	0.7641	0.7844	0.7839
20News	0.6235•	0.7019	0.6395•	0.7049

number of labeled documents $T_l = 64$

dataset	MNB	MNB-EM	SFE	MNB-FM	MNB-WSC
Kitc.	0.5960•	0.5759•	0.6185	0.6040	0.6417
Elec.	0.5905•	0.5606•	0.6129	0.6099	0.6374
T&G	0.5907•	0.5225•	0.6182	0.6135	0.6446
Dvd	0.5329	0.4827•	0.5482	0.5135	0.5546

number of labeled documents $T_l = 128$

dataset	MNB	MNB-EM	SFE	MNB-FM	MNB-WSC
Kitc.	0.6185•	0.5940•	0.6405	0.6308	0.6681
Elec.	0.6117•	0.5829•	0.6346	0.6231•	0.6627
T&G	0.6217•	0.5897•	0.6473•	0.6330•	0.6844
Dvd	0.5259•	0.4880•	0.5607	0.5193•	0.5714

number of labeled documents $T_l = 256$

dataset	MNB	MNB-EM	SFE	MNB-FM	MNB-WSC
Kitc.	0.6431•	0.6001•	0.6639•	0.6528•	0.6955
Elec.	0.6412•	0.5859•	0.6602•	0.6498•	0.6947
T&G	0.6618•	0.6766	0.6726•	0.6533•	0.7156
Dvd	0.5518	0.4937•	0.5787	0.5413•	0.5796

number of labeled documents $T_l = 512$

dataset	MNB	MNB-EM	SFE	MNB-FM	MNB-WSC
Kitc.	0.6777•	0.6158•	0.6904•	0.6787•	0.7182
Elec.	0.6798•	0.5989•	0.6918•	0.6841•	0.7212
T&G	0.6856•	0.7036•	0.6940•	0.6817•	0.7340
Dvd	0.5694•	0.5050•	0.6009	0.5554•	0.6172

Experiments

Our approach can improve the word class distribution estimates for most words.

	Avg Improvement v.s. MNB			Avg Improvement v.s. MNB-EM			Probability Mass		
Word Prop.	Known	Half-Known	Unknown	Known	Half-Known	Unknown	Known	Half-Known	Unknown
$0\text{-}10^{-6}$	-	-0.0658	-0.0607	-	**0.1339**	**0.2206**	-	0.02%	2.11%
$10^{-6}\text{-}10^{-5}$	**0.1919**	**0.0162**	-0.0753	**0.0210**	**0.1474**	**0.1795**	0.03%	1.30%	15.92%
$10^{-5}\text{-}10^{-4}$	**0.0427**	**0.0686**	**0.0289**	**0.0902**	**0.0867**	**0.0926**	2.92%	16.45%	23.25%
$10^{-4}\text{-}10^{-3}$	**0.0579**	**0.0695**	-0.0123	**0.0449**	**0.0618**	**0.3101**	20.67%	12.81%	1.09%
$> 10^{-3}$	-	-	-	-	-	-	-	-	-

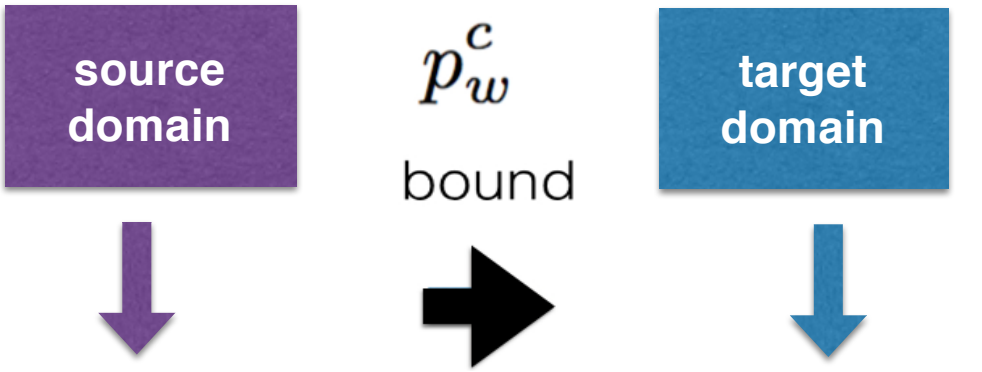
Table 7: Analysis of $\{p_w^+, p_w^-\}$ estimation improvement of MNB-WSC over MNB and MNB-EM(Dataset "Kitchen", $|T_l| = \{64\}$). Known indicates words occurring in both positive and negative training examples, Half Known indicates words occurring in only positive or negative training examples, while Unknown indicates words that never occur in labeled examples.

Transfer Learning for Text Classification with Word-level Statistical Constraint

Word Marginal Distribution Difference $P_s(w) \neq P_t(w)$

Word Class Distribution Consistency $P_s(c|w) = P_t(c|w)$

Electronics	Video Games
(1) Compact; easy to operate; very good picture quality; looks sharp!	(2) A very good game! It is action packed and full of excitement. I am very much hooked on this game.
(3) I purchased this unit from Circuit City and I was very excited about the quality of the picture. It is really nice and sharp.	(4) Very realistic shooting action and good plots. We played this and were hooked.
(5) It is also quite blurry in very dark settings. I will never_buy HP again.	(6) The game is so boring. I am extremely unhappy and will probably never_buy UbiSoft again.



Constraints could be wrong when $P_t(w) >> P_s(w)$!!!

Transfer Learning for Text Classification with Word-level Statistical Constraint

X_w=1 indicates a document of containing w is positive

$$Pr(X_w) = p_w^{+X_w} * (1 - p_w^+)^{1-X_w}, X_w \in \{0, 1\}$$

$$X_w \sim Bernoulli(p_w^+)$$

How to estimate P_w^+? $$P_w^+ = \frac{X_w^1 + X_w^2 + \cdots + X_w^N}{N_w}$$

Maths $$Y = \sum_{k=1}^{n} X_k \sim \mathrm{B}(n,p)$$ (binomial distribution)

When n is large enough

$$(p_w^+)_s \sim N(\mu_s, \frac{\sigma_s^2}{(N_w)_s}) \qquad \mu_s = \frac{(N_w^+)_s}{(N_w^+)_s + (N_w^-)_s} \qquad \sigma_s^2 = \mu_s(1 - \mu_s)$$

Transfer Learning for Text Classification with Word-level Statistical Constraint

When n is large enough

$$(p_w^+)_s \sim N(\mu_s, \frac{\sigma_s^2}{(N_w)_s}) \qquad \mu_s = \frac{(N_w^+)_s}{(N_w^+)_s + (N_w^-)_s} \qquad \sigma_s^2 = \mu_s(1-\mu_s)$$

Similarly for target domain

$$(p_w^+)_t \sim N(\mu_t, \frac{\sigma_t^2}{(N_w)_t}) \qquad \mu_t = \frac{\sum_{d \in T} N_{d,w} \times P(+|d)}{(N_w)_t} \qquad \sigma_t^2 = \mu_t(1-\mu_t)$$

Transfer Learning for Text Classification with Word-level Statistical Constraint

Idea: calculate confidence interval for $(p_w^+)_s - (p_w^+)_t$, considering the sample variance in both source domain and target domain.

$$(p_w^+)_s - (p_w^+)_t \sim N(\mu, \sigma^2)$$

$$\mu = \mu_s - \mu_t$$

$$\sigma^2 = \frac{\sigma_s^2}{(N_w)_s} + \frac{\sigma_t^2}{(N_w)_t}$$

$$\mu - z_{\alpha/2} \times \sigma \leq (p_w^+)_s - (p_w^+)_t \leq \mu + z_{\alpha/2} \times \sigma$$

According to the assumption

$$(p_w^+)_s - (p_w^+)_t = 0$$

$$\mu - z_{\alpha/2} \times \sigma \leq 0 \leq \mu + z_{\alpha/2} \times \sigma$$

Transfer Learning for Text Classification with Word-level Statistical Constraint

Idea: calculate confidence interval for $(p_w^+)_s - (p_w^+)_t$, considering the sample variance in both source domain and target domain.

$$\frac{-b-\sqrt{b^2-4ac}}{2a} \leq \mu_t \leq \frac{-b+\sqrt{b^2-4ac}}{2a}$$

$$a = 1 + z^2_{\alpha/2}\frac{1}{N_w}$$

$$b = -(2\mu_s + z^2_{\alpha/2}\frac{1}{(N_w)_t})$$

$$c = \mu_s^2 - z^2_{\alpha/2}\frac{\sigma_s^2}{(N_w)_s}$$

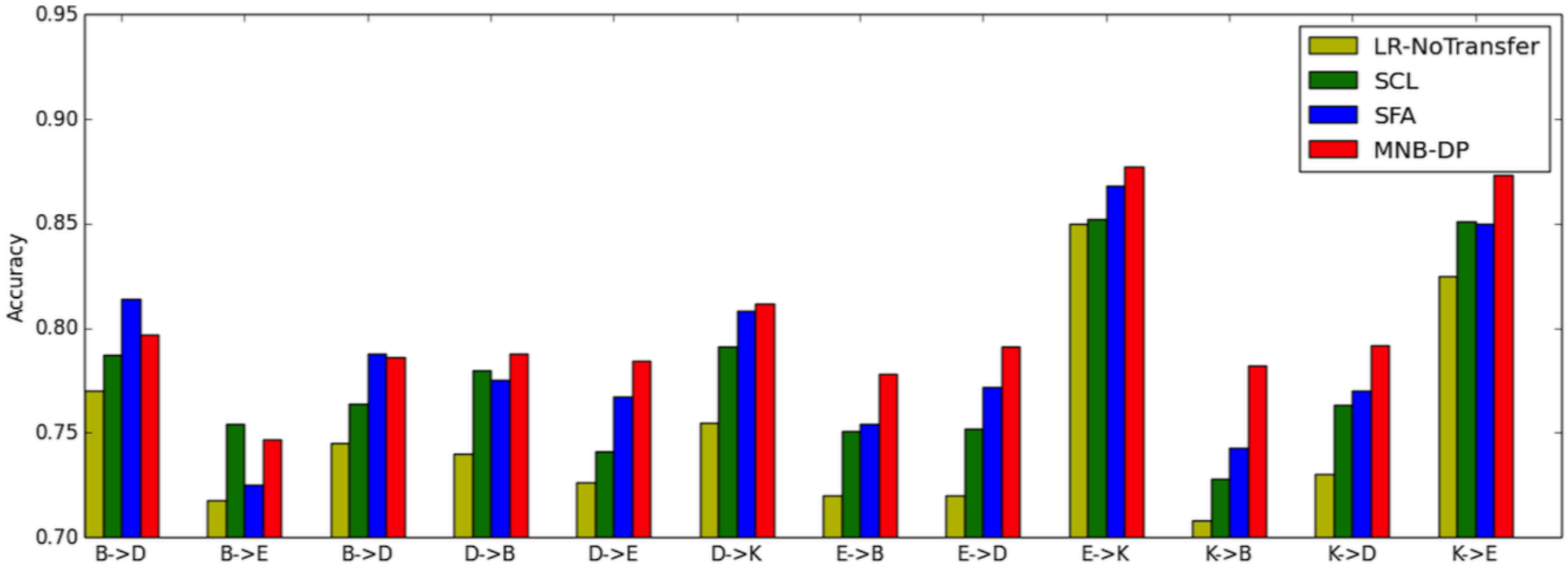
Constraints on document posteriors

$$Q = \{q(c|d)| \sum_{d\in U} N_{d,w} \times q(c|d) \leq upper(\mu_t) \times (N_w)_u,$$

$$\sum_{d\in U} N_{d,w} \times q(c|d) \geq lower(\mu_t) \times (N_w)_u\}$$

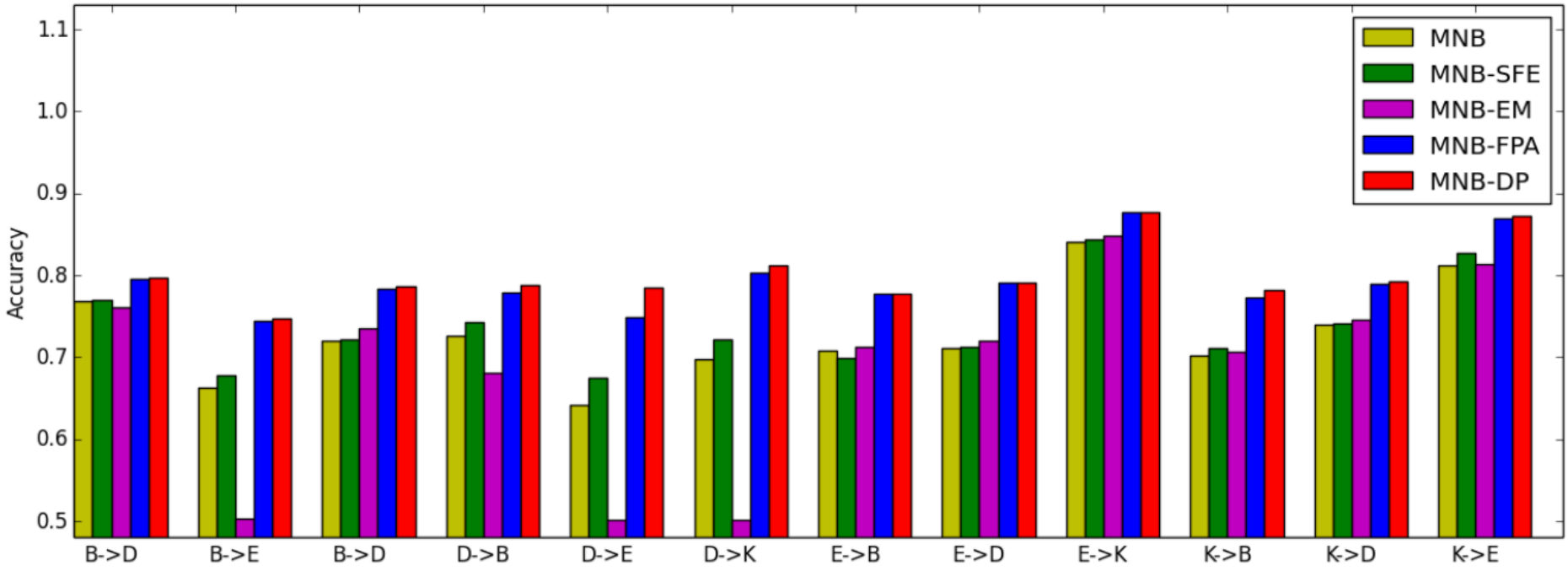
Transfer Learning Experiments

Dataset	#Reviews	#Pos	#Neg	#Features
dvds	2,000	1,000	1,000	473,856
kitchen	2,000	1,000	1,000	
electronics	2,000	1,000	1,000	
books	2,000	1,000	1,000	



Transfer Learning Experiments

Dataset	#Reviews	#Pos	#Neg	#Features
dvds	2,000	1,000	1,000	473,856
kitchen	2,000	1,000	1,000	
electronics	2,000	1,000	1,000	
books	2,000	1,000	1,000	



Summary of PR

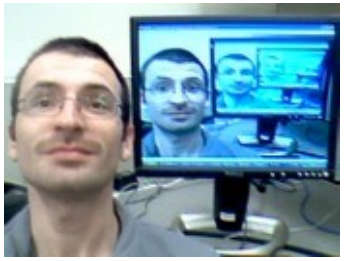
1. Design the constraints (and how to apply it robustly)
2. Formulate the problem within PR by designing
    1. Probabilistic model
    2. Data likelihood
    3. The KL term
3. Solve the problem

Constraint Driven Learning

Constraint-Driven Learning

M. Chang, L. Ratinov, D. Roth (2007).

University of Illinois at Urbana-Champaign (2007)


Application: Information Extraction

Idea: Tell the system~~~

- Citations have contiguous authors
- Citation fields usually end with punctuation

Implementation:

- Design a penalty function to encode constraint

Constraint-Driven Learning

M. Chang, L. Ratinov, D. Roth (2007).

University of Illinois at Urbana-Champaign (2007)

Idea: Use knowledge to decode better:

predict 25% of articles are "politics"

$$\hat{\mathbf{Y}} = \arg\max_{\mathbf{Y}} \quad \log p_\theta(\mathbf{Y}|\mathbf{X}) - \text{penalty}(\mathbf{Y})$$

Constraint-Driven Learning

M. Chang, L. Ratinov, D. Roth (2007).

University of Illinois at Urbana-Champaign (2007)

Idea: Use knowledge to decode better:

predict 25% of articles are "politics"

Idea: Retrain with predictions.

Constraint Driven Learning:

$$\text{E-Step: set } \hat{\mathbf{Y}} = \arg\max_{\mathbf{Y}} \log p_\theta(\mathbf{Y}|\mathbf{X}) - \text{penalty}(\mathbf{Y})$$

$$\text{M-Step: set } \theta = \arg\max_{\theta} \log p_\theta(\hat{\mathbf{Y}}|\mathbf{X})$$

Constraint-Driven Learning

M. Chang, L. Ratinov, D. Roth (2007).

Motivation: Hard EM-like algorithm with preferences

Constraint Driven Learning:

$$\text{E-Step: set } \hat{\mathbf{Y}} = \arg\max_{\mathbf{Y}} \log p_\theta(\mathbf{Y}|\mathbf{X}) - \text{penalty}(\mathbf{Y})$$

$$\text{M-Step: set } \theta = \arg\max_{\theta} \log p_\theta(\hat{\mathbf{Y}}|\mathbf{X})$$

- penalties encode similar information as $\mathbf{E}[\phi] \approx \mathbf{b}$

$$\text{penalty}(\mathbf{Y}) = ||\phi(\mathbf{X}, \mathbf{Y}) - \mathbf{b}||_\beta$$

- E-Step can be hard; use beam search

Constraint-Driven Learning

M. Chang, L. Ratinov, D. Roth (2007).

Input:

$Cycles$: learning cycles
$Tr = \{x, y\}$: labeled training set.
U: unlabeled dataset
F: set of feature functions.
$\{\rho_i\}$: set of penalties.
$\{C_i\}$: set of constraints.
γ: balancing parameter with the supervised model.
$learn(Tr, F)$: supervised learning algorithm
Top-K-Inference:
returns top-K labeled scored by the cost function (1)

CODL:

1. Initialize $\boldsymbol{\lambda}_0 = learn(Tr, F)$.
2. $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$.
3. For $Cycles$ iterations do:
4. $T = \phi$
5. For each $x \in U$
6. $\{(x, y^1), \ldots, (x, y^K)\} =$
7. Top-K-Inference$(x, \boldsymbol{\lambda}, F, \{C_i\}, \{\rho_i\})$
8. $T = T \cup \{(x, y^1), \ldots, (x, y^K)\}$
9. $\boldsymbol{\lambda} = \gamma\boldsymbol{\lambda}_0 + (1 - \gamma)learn(T, F)$

Train a learning model

Constraint-Driven Learning

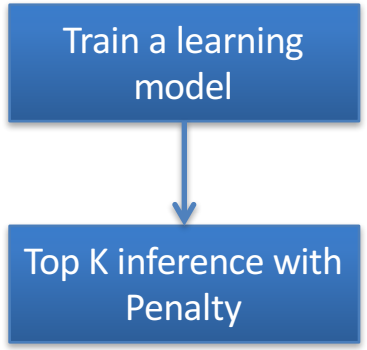
M. Chang, L. Ratinov, D. Roth (2007).

Input:

$Cycles$: learning cycles
$Tr = \{x, y\}$: labeled training set.
U: unlabeled dataset
F: set of feature functions.
$\{\rho_i\}$: set of penalties.
$\{C_i\}$: set of constraints.
γ: balancing parameter with the supervised model.
$learn(Tr, F)$: supervised learning algorit
Top-K-Inference:
returns top-K labeled scored by the cos

CODL:

1. Initialize $\boldsymbol{\lambda}_0 = learn(Tr, F)$.
2. $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$.
3. For $Cycles$ iterations do:
4. $T = \phi$
5. For each $x \in U$
6. $\{(x, y^1), \ldots, (x, y^K)\} =$
7. Top-K-Inference$(x, \boldsymbol{\lambda}, F, \{C_i\}, \{\rho_i\})$
8. $T = T \cup \{(x, y^1), \ldots, (x, y^K)\}$
9. $\boldsymbol{\lambda} = \gamma\boldsymbol{\lambda}_0 + (1 - \gamma)learn(T, F)$



$$\underset{y}{\operatorname{argmax}}\, \boldsymbol{\lambda} \cdot F(x, y) - \sum_{i=1}^{K} \rho_i d(y, 1_{C_i(x)})$$

Constraint-Driven Learning

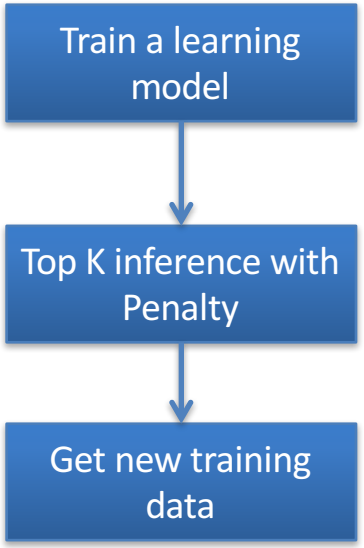
M. Chang, L. Ratinov, D. Roth (2007).

Input:

$Cycles$: learning cycles
$Tr = \{x, y\}$: labeled training set.
U: unlabeled dataset
F: set of feature functions.
$\{\rho_i\}$: set of penalties.
$\{C_i\}$: set of constraints.
γ: balancing parameter with the supervised model.
$learn(Tr, F)$: supervised learning algorithm
Top-K-Inference:
returns top-K labeled scored by the cost function (1)

CODL:

1. Initialize $\boldsymbol{\lambda}_0 = learn(Tr, F)$.
2. $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$.
3. For $Cycles$ iterations do:
4. $T = \phi$
5. For each $x \in U$
6. $\{(x, y^1), \ldots, (x, y^K)\} =$
7. Top-K-Inference$(x, \boldsymbol{\lambda}, F, \{C_i\}, \{\rho_i\})$
8. $T = T \cup \{(x, y^1), \ldots, (x, y^K)\}$
9. $\boldsymbol{\lambda} = \gamma\boldsymbol{\lambda}_0 + (1 - \gamma)learn(T, F)$



Constraint-Driven Learning

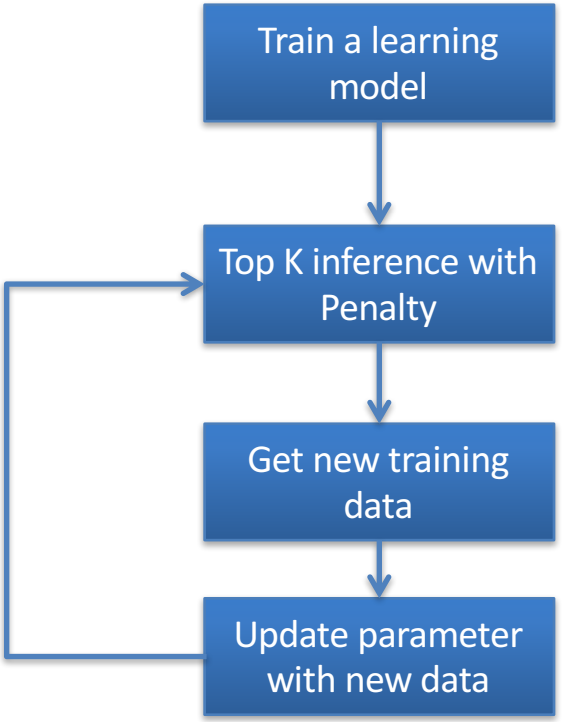
M. Chang, L. Ratinov, D. Roth (2007).

Input:

$Cycles$: learning cycles
$Tr = \{x, y\}$: labeled training set.
U: unlabeled dataset
F: set of feature functions.
$\{\rho_i\}$: set of penalties.
$\{C_i\}$: set of constraints.
γ: balancing parameter with the supervised model.
$learn(Tr, F)$: supervised learning algorithm
Top-K-Inference:
returns top-K labeled scored by the cost function (1)

CODL:

1. Initialize $\boldsymbol{\lambda}_0 = learn(Tr, F)$.
2. $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$.
3. For $Cycles$ iterations do:
4. $T = \phi$
5. For each $x \in U$
6. $\{(x, y^1), \ldots, (x, y^K)\} =$
7. Top-K-Inference$(x, \boldsymbol{\lambda}, F, \{C_i\}, \{\rho_i\})$
8. $T = T \cup \{(x, y^1), \ldots, (x, y^K)\}$
9. $\boldsymbol{\lambda} = \gamma\boldsymbol{\lambda}_0 + (1 - \gamma)learn(T, F)$



Generalized Expectation Criterion

Generalized Expectation Constraints

G. Mann, A. McCallum (2007). G. Druck, G. Mann, A. McCallum. (2008)

University of Massachusetts Amherst (2007)



Application: Document Classification, Info Extraction

Idea: Use labeled features:

- Document has $\text{“puck”} \Rightarrow p(\text{class} = \text{sport}) = 90\%$

Implementation:

- Add penalty while training:

$$\max_{\theta} \mathcal{L}_{\theta} \Longrightarrow \max_{\theta} \mathcal{L}_{\theta} + \text{penalty}(p_{\theta}(\mathbf{Y}|\mathbf{X}))$$

Generalized Expectation Constraints

G. Mann, A. McCallum (2007). G. Druck, G. Mann, A. McCallum. (2008)

University of Massachusetts Amherst (2007)

Idea: Penalize "bad" distributions:
train a model to predict 25% of articles as "politics"

Generalized Expectation Constraints

G. Mann, A. McCallum (2007). G. Druck, G. Mann, A. McCallum. (2008)

University of Massachusetts Amherst (2007)

Idea: Penalize "bad" distributions:
train a model to predict 25% of articles as "politics"

Objective:

$$\max_{\theta} \mathcal{L}(\theta; D_L) - \left\| \mathbf{E}_{p_\theta(\mathbf{Y}|\mathbf{X})}[\phi] - \mathbf{b} \right\|_\beta$$

where $\mathbf{E}_{p_\theta(\mathbf{Y}|\mathbf{X})}[\phi] = \mathbf{E}_{p_\theta(\mathbf{Y}|\mathbf{X})}[\phi(\mathbf{X}, \mathbf{Y})]$

$$= \sum_{\mathbf{Y}} p_\theta(\mathbf{Y}|\mathbf{X})\phi(\mathbf{X}, \mathbf{Y})$$ is short-hand

Optimization: gradient descent on θ

Generalized Expectation Constraints

G. Druck, G. Mann, A. McCallum. (2008)

Objective: measure the *distance* between a reference expectation and predicted expectation for features ***f(x,y)***

$$\Delta(\hat{f}, E_{\tilde{p}(X)}[E_{p_\theta(Y|X)}[f(X,Y)]])$$

$$-\sum_{k \in K} D(\hat{p}(y|x_k > 0))||\tilde{p}_\theta(y|x_k > 0)) - \sum_j \frac{\theta_j^2}{2\sigma^2}$$

KL [

	Positive	Negative
love	0.9	0.1
interest	0.9	0.1
bad	0.1	0.9
hate	0.1	0.9

Reference distribution

||

	positive	Negative
love	0.7	0.3
interest	0.6	0.4
bad	0.2	0.8
hate	0.1	0.9

Predicted distribution

]

A Real Application for GEC
What if we have no data?

Text Classification
What if we have no data?

but the majority of the film is a convoluted and confusing **mess** . characters keep popping up with no explanation , demanding money for deals that occur off-screen

but what in the end makes " toy story 2 " a **memorable** experience is not the jokes , its multiple parodies or **marvelous** animation . it is its heart and emotions

Objective: $\mathcal{L} = \sum_{(x) \in D} \log(\sum_{y} p_\theta(y|x)) = 0$

Cannot use standard unsupervised learning with ME

We still have some prior knowledge about the problem

Text Classification
What if we have no data?

but the majority of the film is a convoluted and confusing **mess** . characters keep popping up with no explanation , demanding money for deals that occur off-screen

but what in the end makes " toy story 2 " a **memorable** experience is not the jokes , its multiple parodies or **marvelous** animation . it is its heart and emotions

Objective: $\mathcal{L} = \sum_{(x) \in D} \log(\sum_{y} p_\theta(y|x)) = 0$

Cannot use standard unsupervised learning with ME

We still have some prior knowledge about the problem

Positive: memorable, marvelous

Negative: mess

Text Classification
Labeled features

[Mann & McCallum 07], [Druck et al. 08]

Text Classification
Labeled features

[Mann & McCallum 07], [Druck et al. 08]

- **feature:**

$$\phi_{w\ell}(\mathbf{x}, y) \begin{cases} 1 & w \in \mathbf{x} \;\&\; y = \ell \\ 0 & otherwise \end{cases}$$

- **expectation:** label distribution for docs that contain *w*

Text Classification
Labeled features

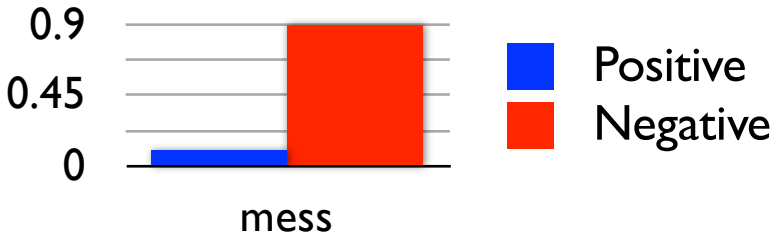
[Mann & McCallum 07], [Druck et al. 08]

- **feature:**

$$\phi_{w\ell}(\mathbf{x}, y) \begin{cases} 1 & w \in \mathbf{x} \;\&\; y = \ell \\ 0 & otherwise \end{cases}$$

- **expectation:** label distribution for docs that contain *w*

$$\frac{1}{c_w} \sum_{\mathbf{x}} \mathrm{E}_{p_\theta(y|\mathbf{x})}[\phi_w(\mathbf{x}, y)]$$



Text Classification
Labeled features

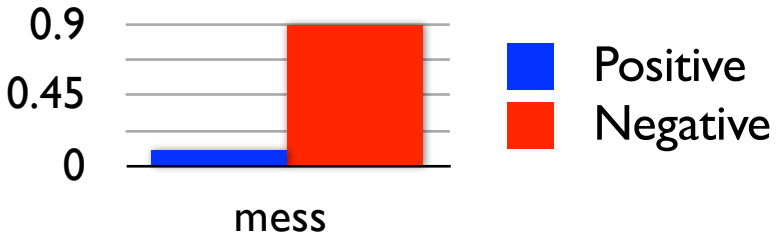
[Mann & McCallum 07], [Druck et al. 08]

- **feature:**

$$\phi_{w\ell}(\mathbf{x}, y) \begin{cases} 1 & w \in \mathbf{x} \;\&\; y = \ell \\ 0 & otherwise \end{cases}$$

- **expectation:** label distribution for docs that contain *w*

$$\frac{1}{c_w} \sum_{\mathbf{x}} \mathrm{E}_{p_\theta(y|\mathbf{x})}[\phi_w(\mathbf{x}, y)]$$



- **GE penalty:** KL divergence from target distribution

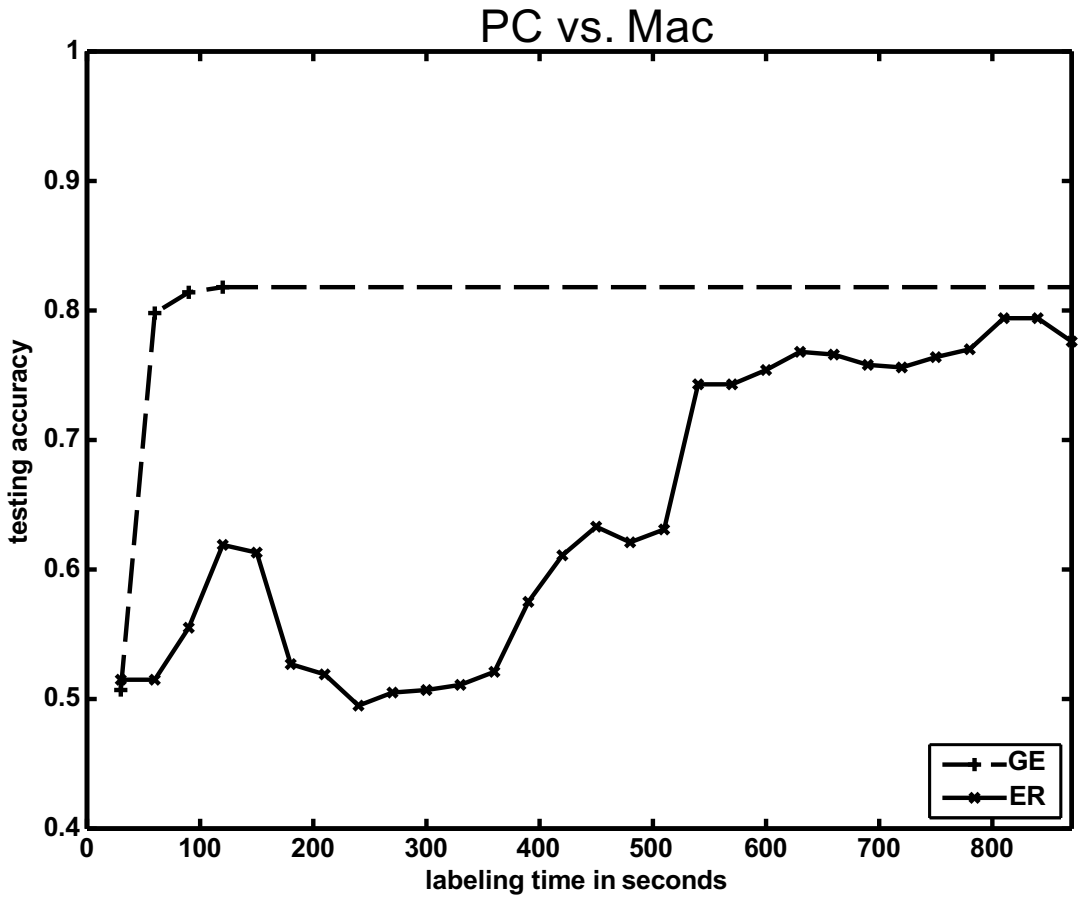
$$\mathcal{D}_{KL}\left(b || \frac{1}{c_w} \sum_{\mathbf{x}} \mathrm{E}_{p_\theta(y|\mathbf{x})}[\phi_w(\mathbf{x}, y)]\right)$$

User Experiments with Labeled Features

[Druck et al. 08]

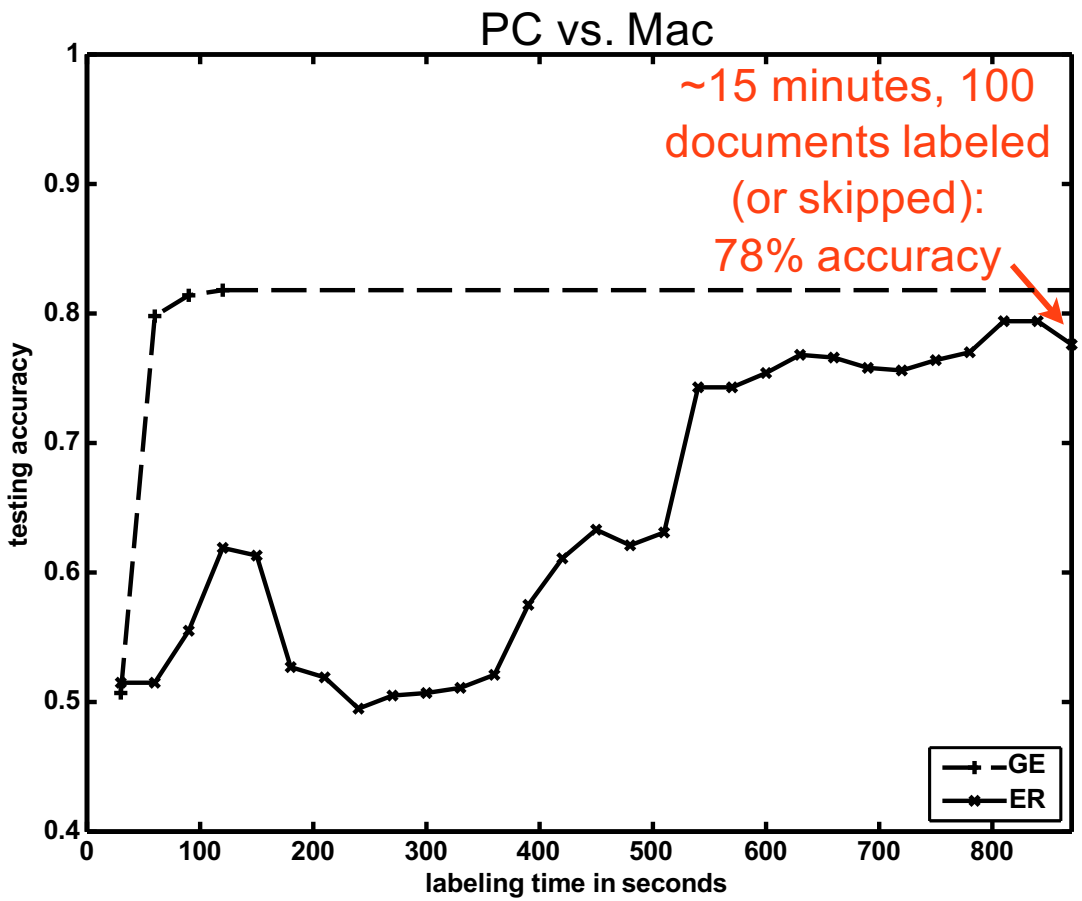
User Experiments with Labeled Features

[Druck et al. 08]



User Experiments with Labeled Features

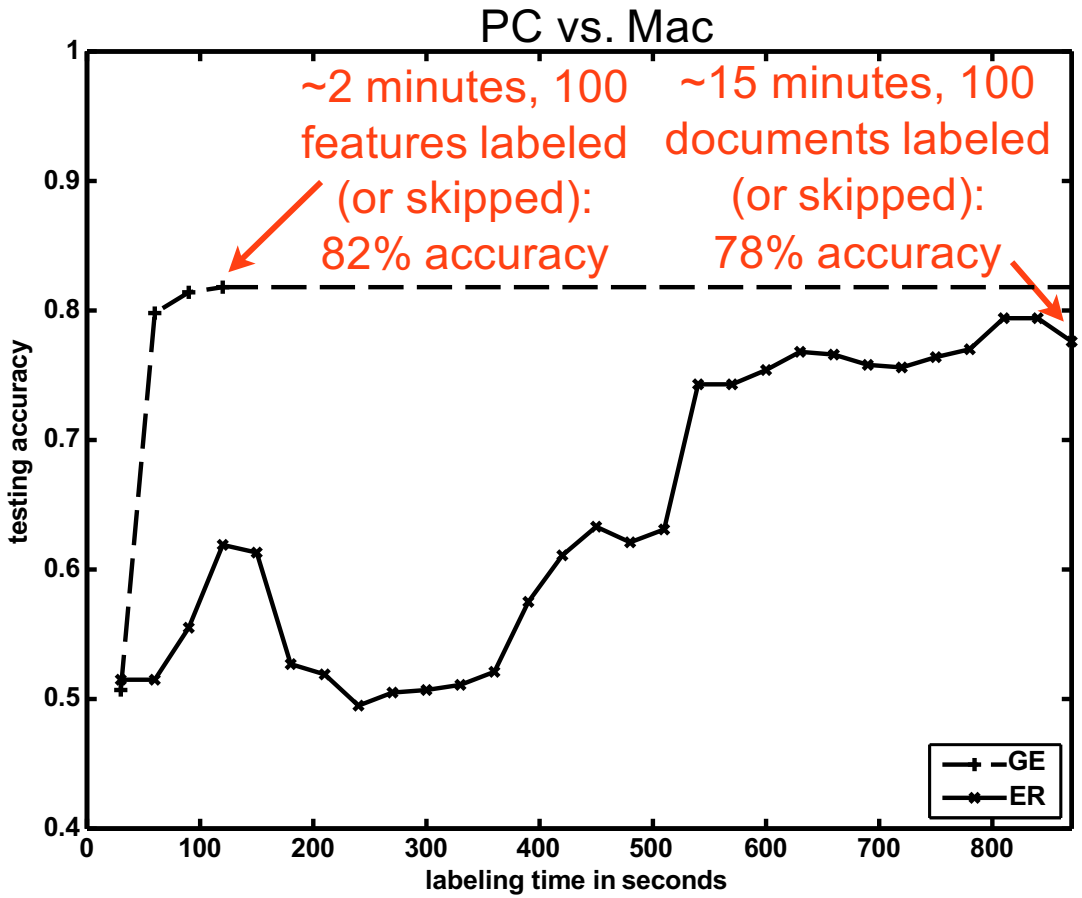
[Druck et al. 08]



updated slides: http://sideinfo.wikkii.com 93

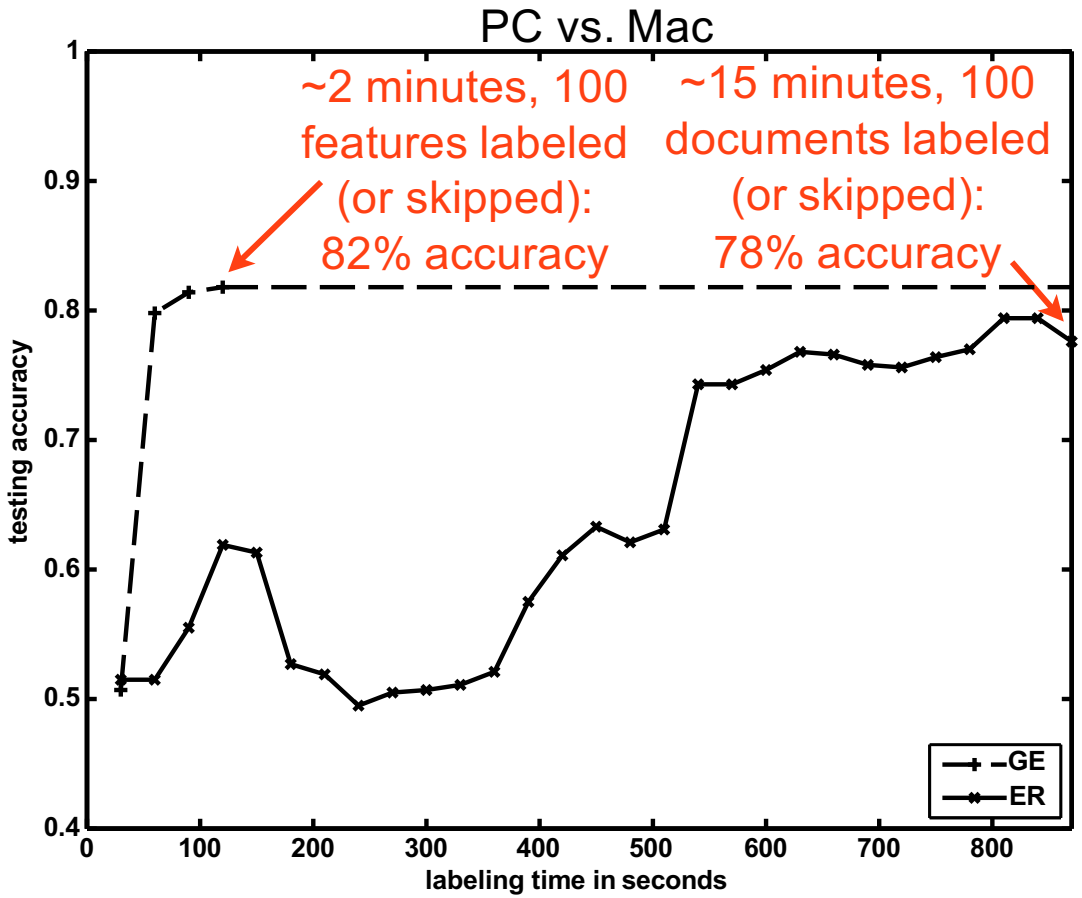
User Experiments with Labeled Features

[Druck et al. 08]



User Experiments with Labeled Features

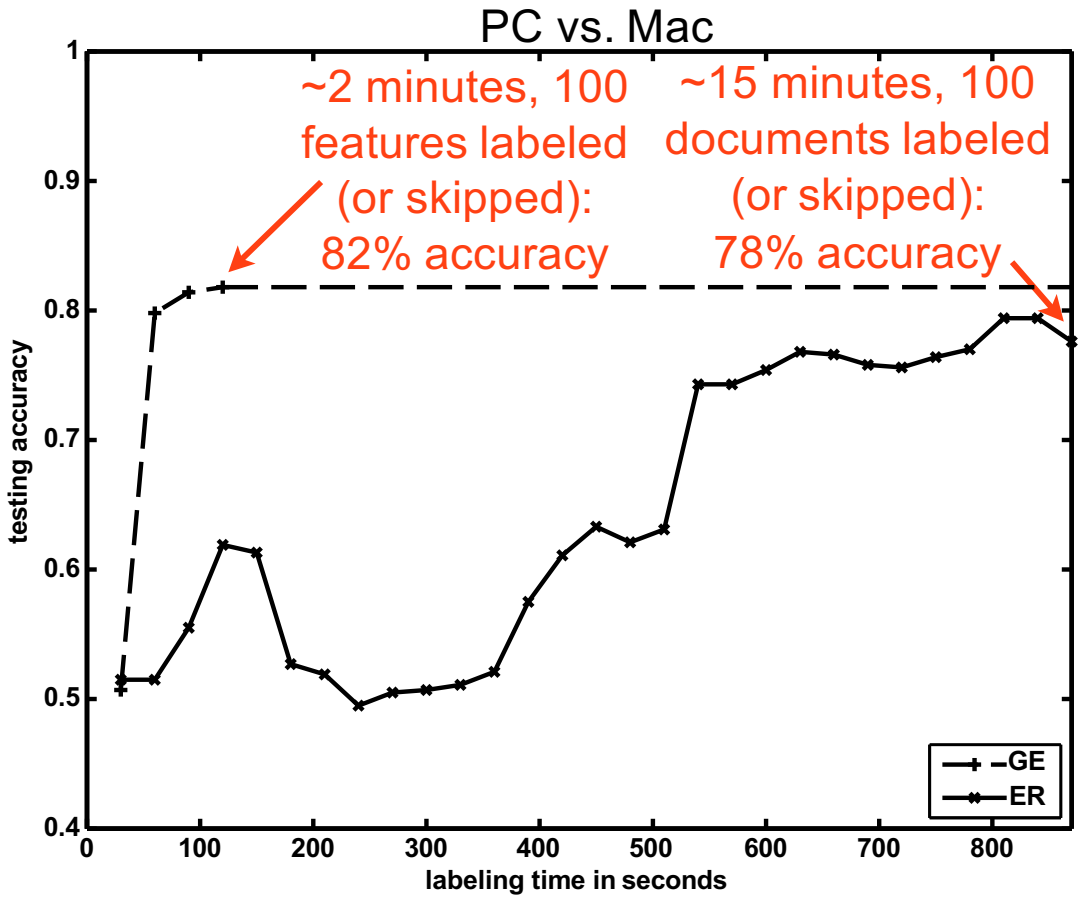
[Druck et al. 08]



targets set with simple heuristic: majority label gets 90% of mass

User Experiments with Labeled Features

[Druck et al. 08]



targets set with simple heuristic: majority label gets 90% of mass

complete set of labeled features

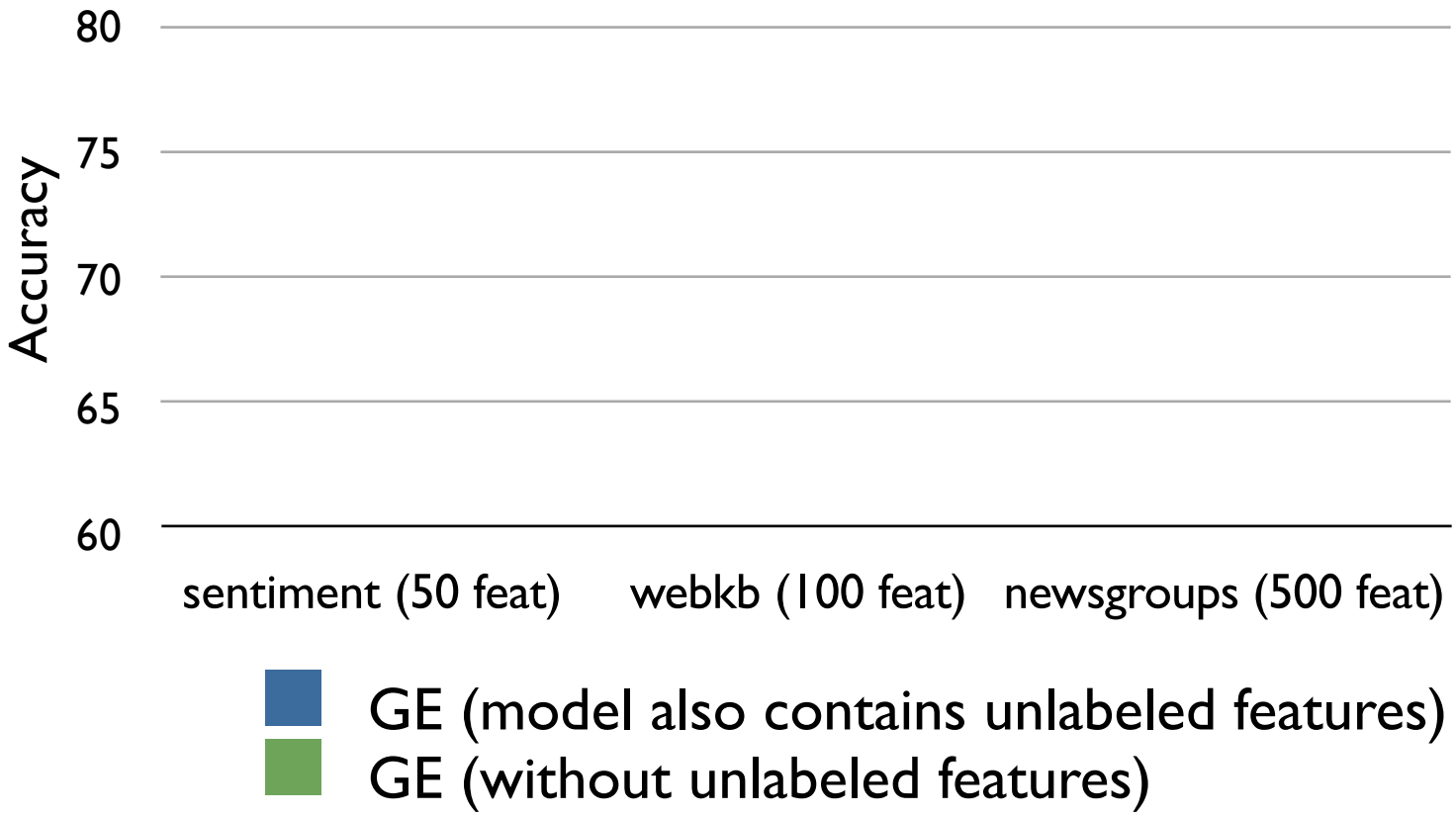
PC	Mac
dos	mac
ibm	apple
hp	quadra
dx	

Experiments with Labeled Features

[Druck et al. 08]

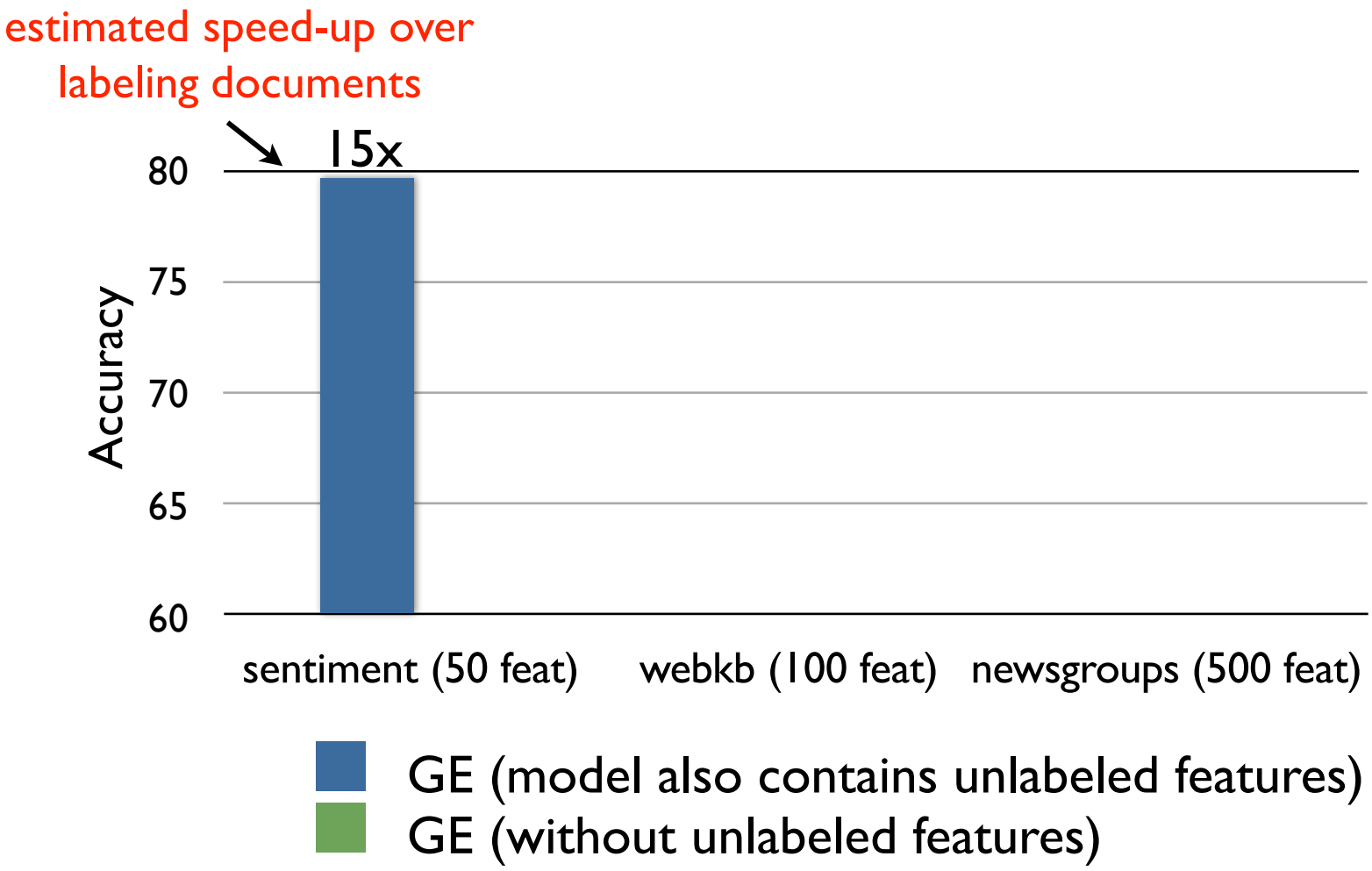
Experiments with Labeled Features

[Druck et al. 08]



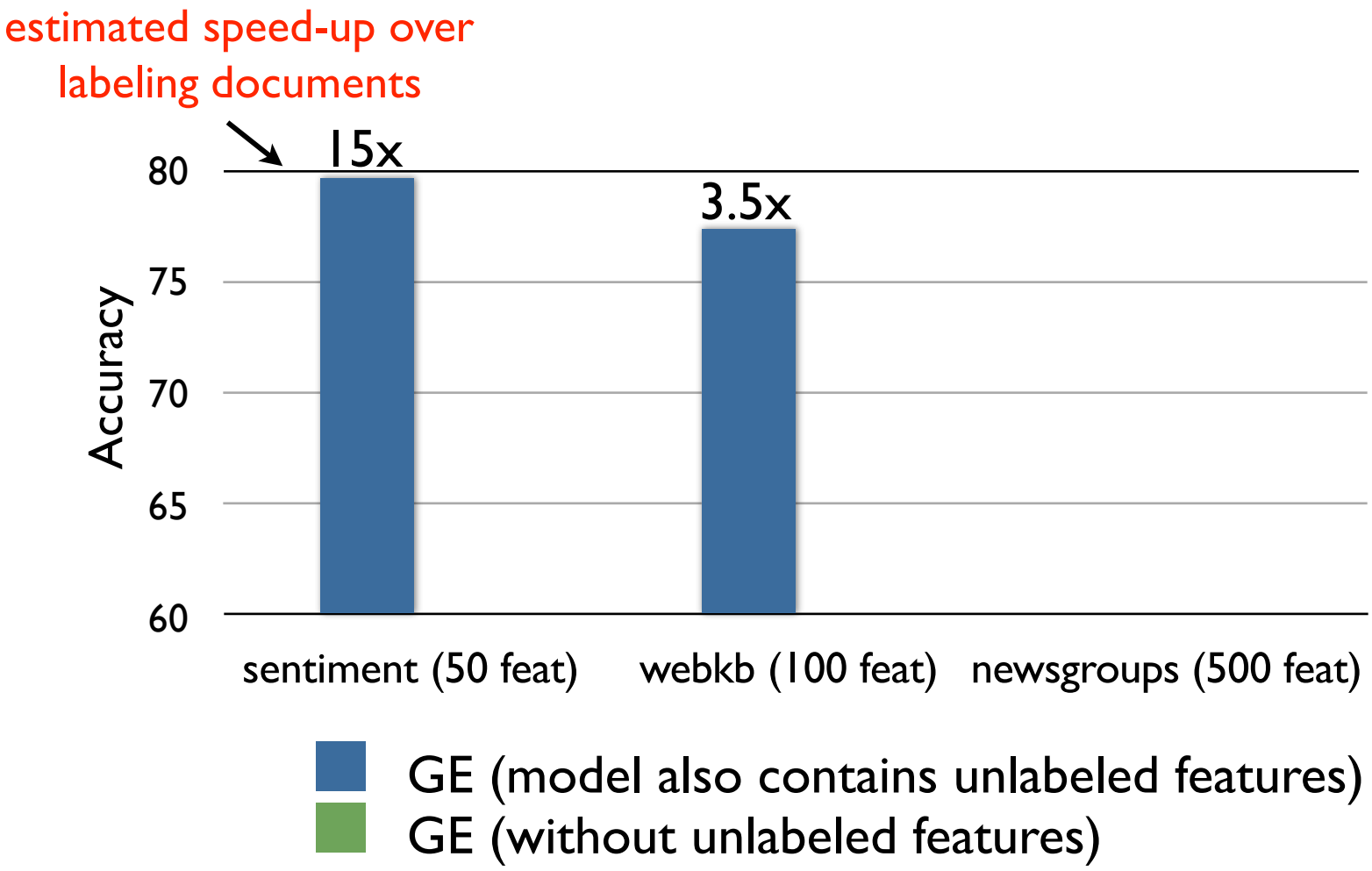
Experiments with Labeled Features

[Druck et al.08]



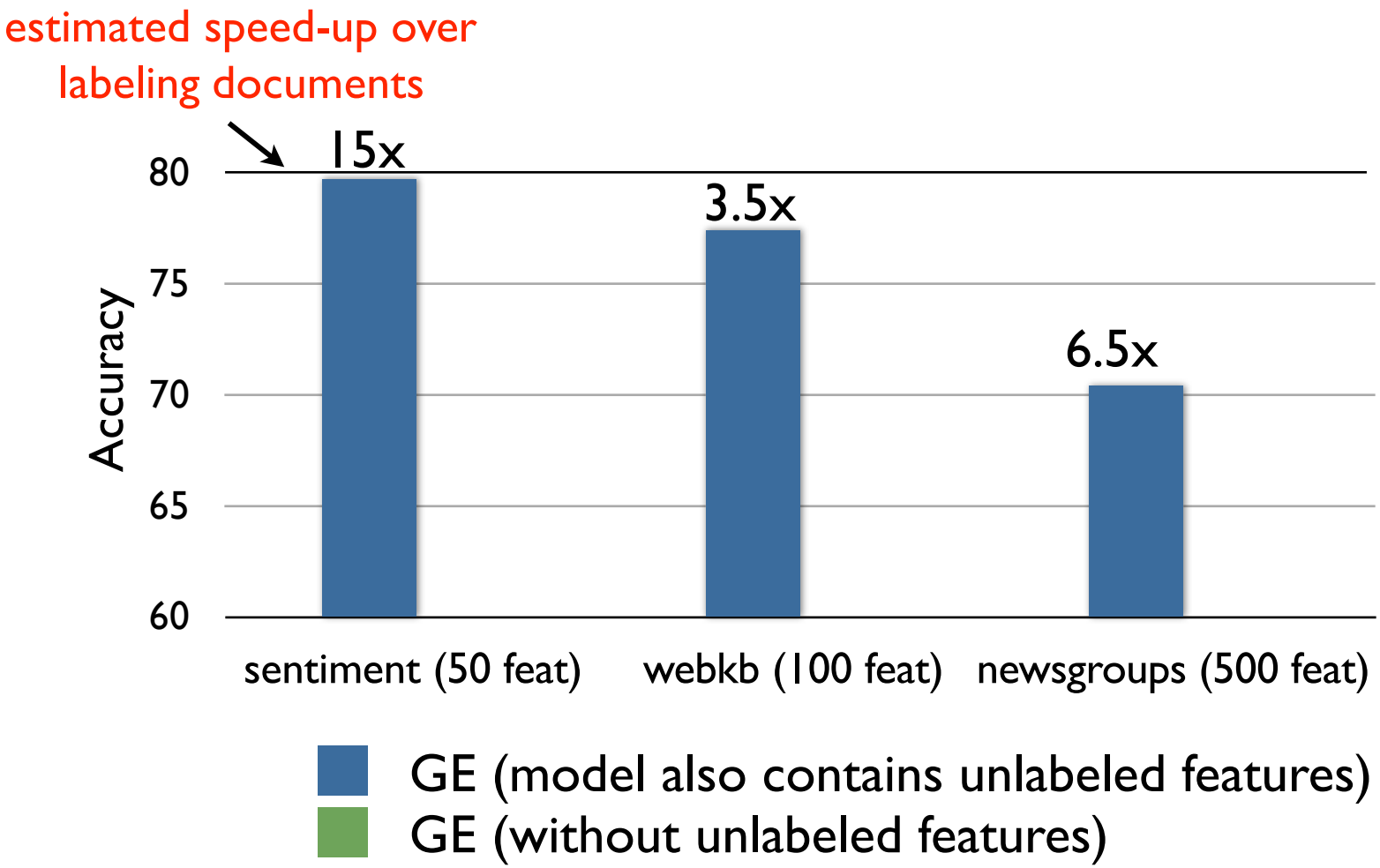
Experiments with Labeled Features

[Druck et al.08]



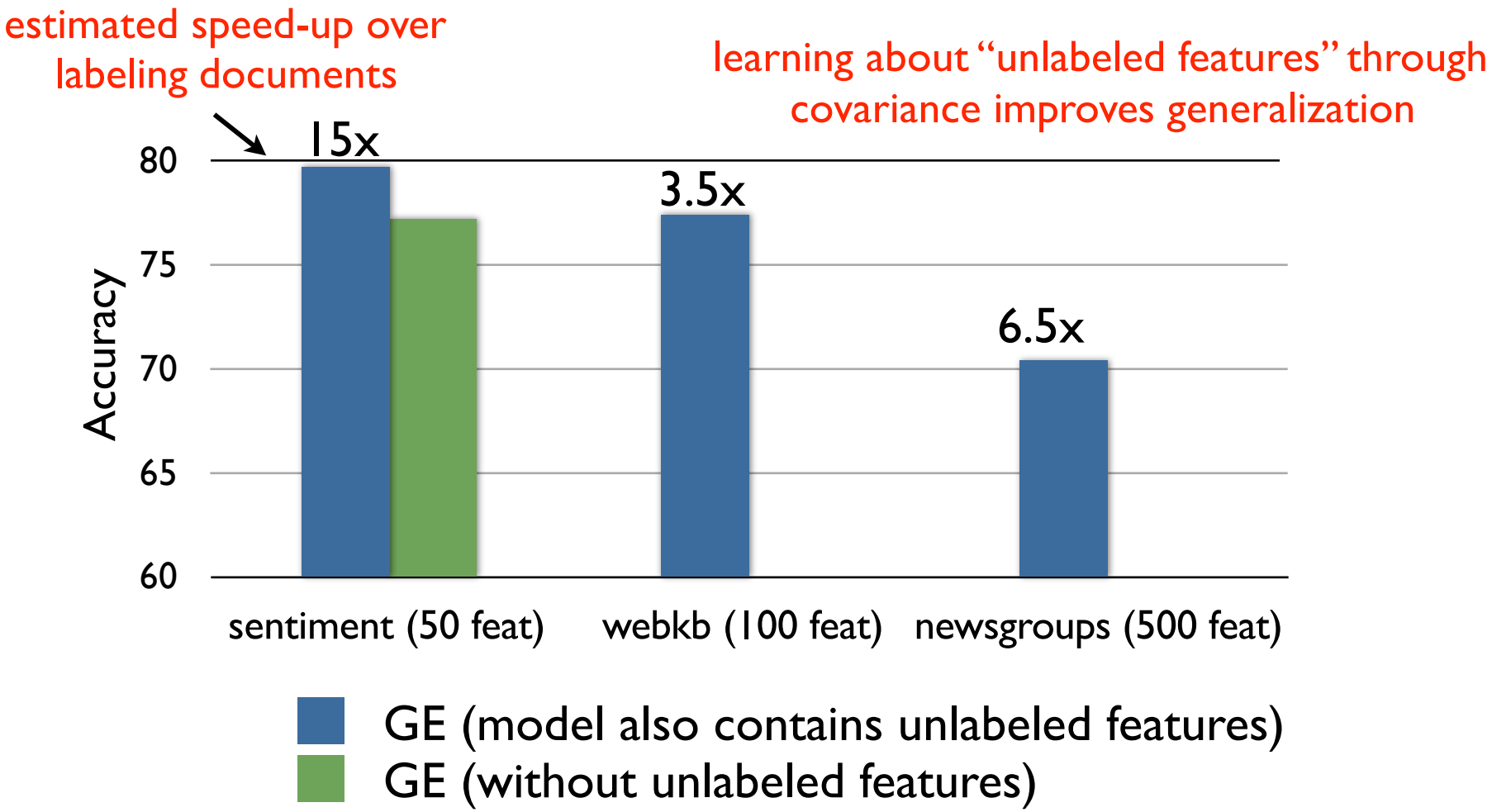
Experiments with Labeled Features

[Druck et al.08]



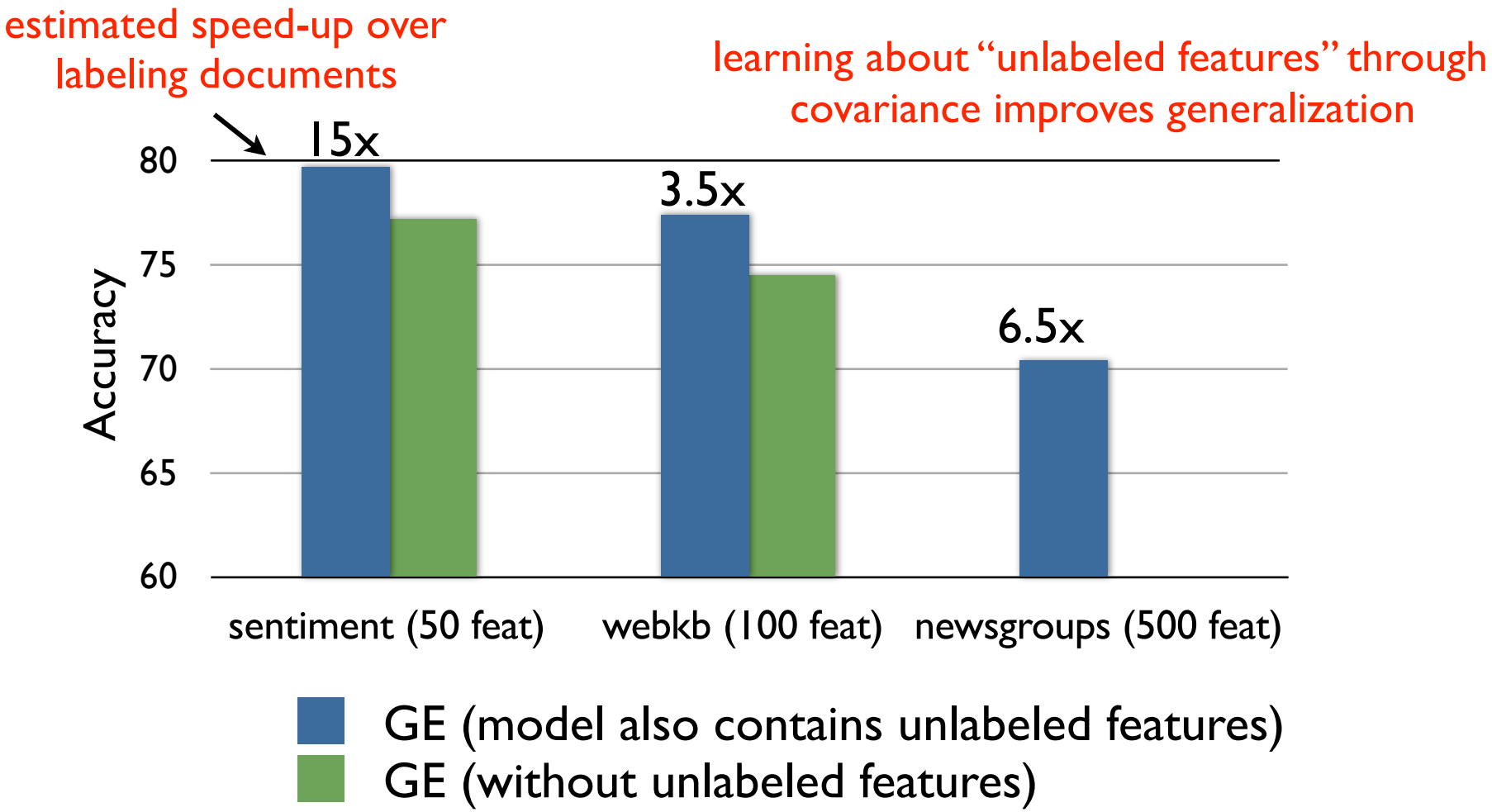
Experiments with Labeled Features

[Druck et al.08]



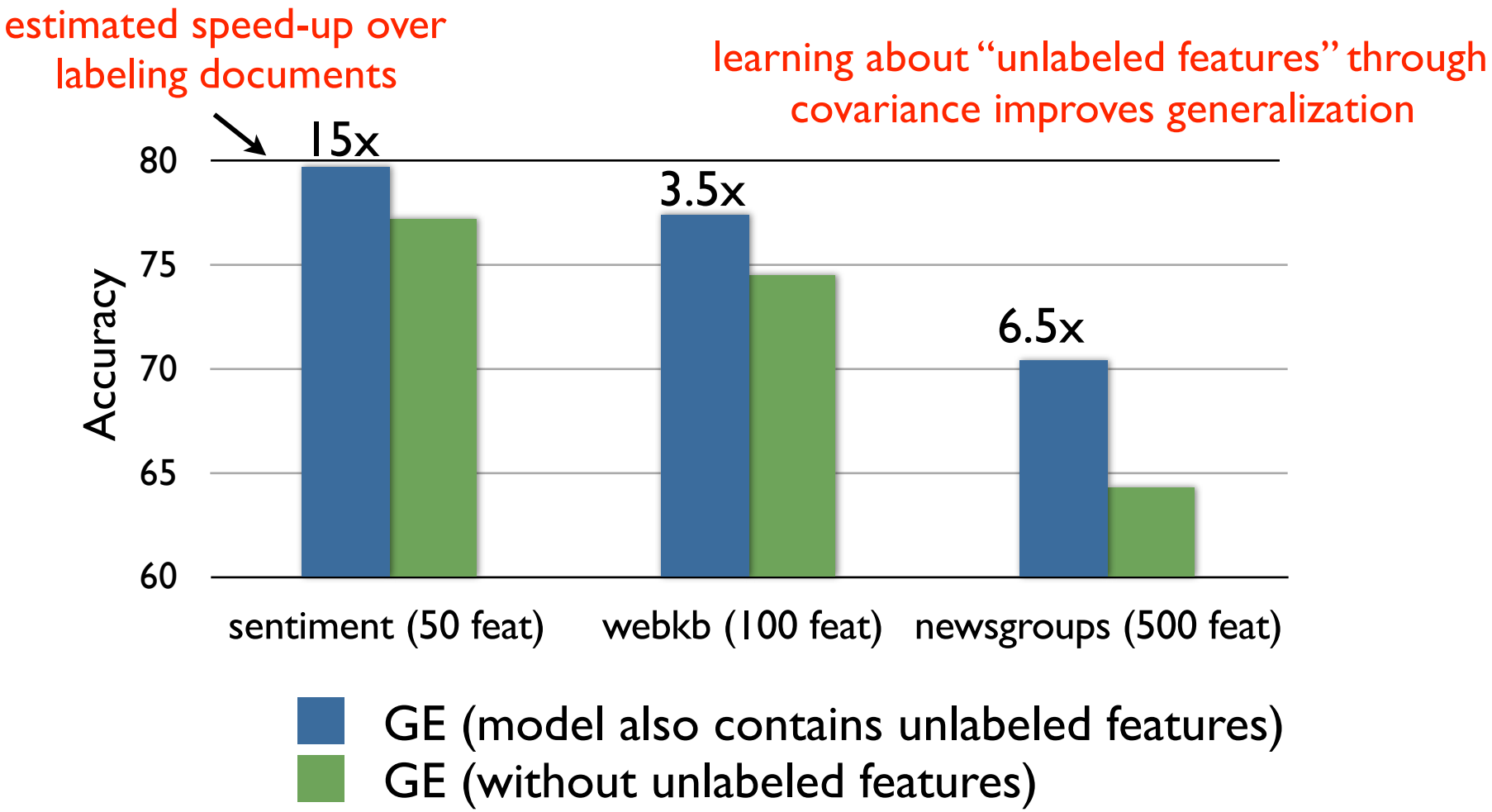
Experiments with Labeled Features

[Druck et al.08]



Experiments with Labeled Features

[Druck et al.08]



API for New GE Constraints: MALLET

http://mallet.cs.umass.edu

- *Java Interfaces* for implementing ***new*** GE constraints
- covariance computation implemented (MaxEnt, CRF)
- ***primarily need to write code to:***
  - *compute constraint features*
  - *compute penalty and penalty-specific part of the gradient*
- ***restriction:*** constraints must factor with model
- ***restriction:*** penalty should be differentiable

Summary: CoDL, GE, PR

Constraint Driven Learning:

Apply constraints at decode time + self-training.

$$\arg\max_{\mathbf{Y}} \quad \log p_\theta(\mathbf{Y}|\mathbf{X}) - \text{penalty}(\mathbf{Y})$$

Generalized Expectation Constraints:

Train model to satisfy constraints.

$$\max_\theta \quad \mathcal{L}_\theta \quad \Longrightarrow \quad \max_\theta \quad \mathcal{L}_\theta - \text{penalty}(p_\theta(\mathbf{Y}|\mathbf{X}))$$

Posterior Regularization:

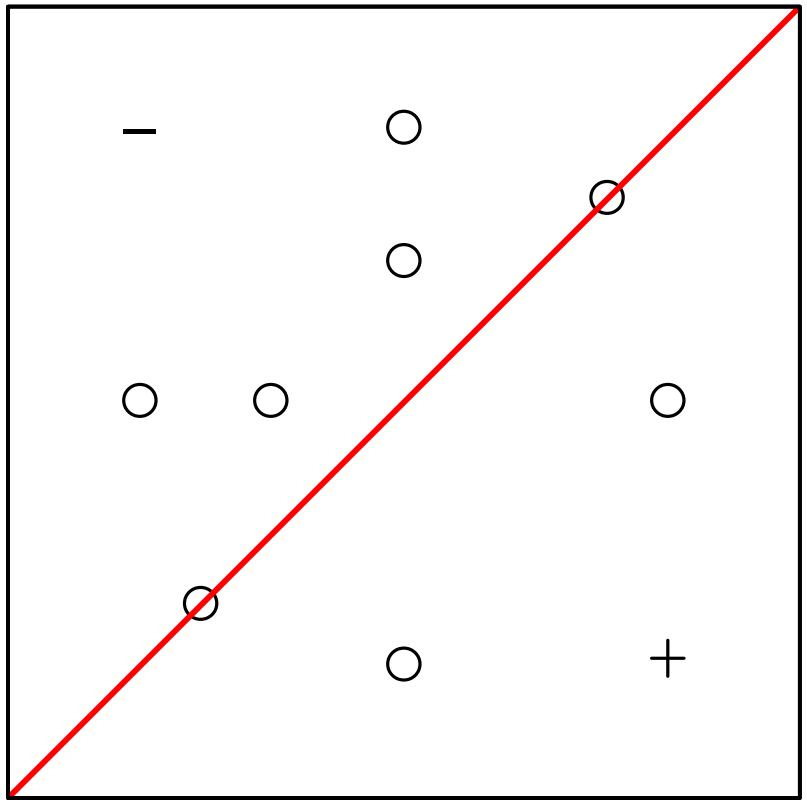
Project onto a constraint set + EM training.

$$\max_\theta \quad \mathcal{L}_\theta \quad \Longrightarrow \quad \max_\theta \quad \mathcal{L}(\theta; D_L) \; - \; \mathcal{D}_{\text{KL}}(\mathcal{Q} || \; p_\theta(\mathbf{Y}|\mathbf{X}))$$

Visual Example: Maximum Likelihood

Model: $p(\mathbf{Y}|\mathbf{X}) = \prod_i \frac{\exp(\mathbf{y}_i \mathbf{x}_i \cdot \theta)}{Z(\mathbf{x}_i)}$

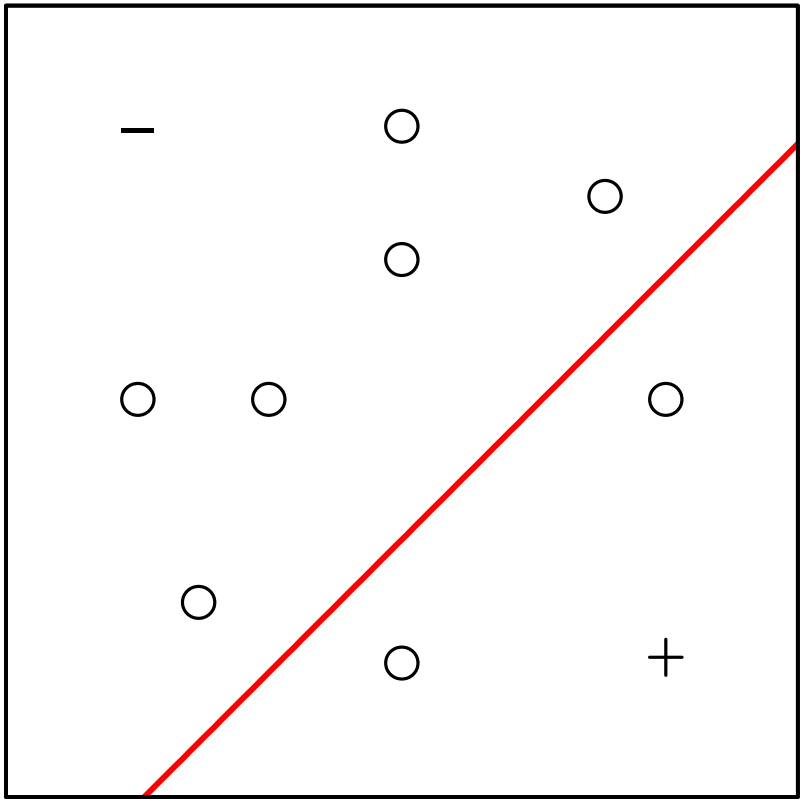
Objective: $\max_\theta \ \log p_\theta(\mathbf{Y}_L|\mathbf{X}_L) - 0.1\|\theta\|_2^2$



Visual Example: Constraint Driven Learning

$$\max_{\theta, \hat{\mathbf{Y}}} \quad \log p_\theta(\mathbf{Y}_L | \mathbf{X}_L) - 0.1\|\theta\|_2^2 \quad \text{s.t.} \quad \phi(\hat{\mathbf{Y}}) = 2$$

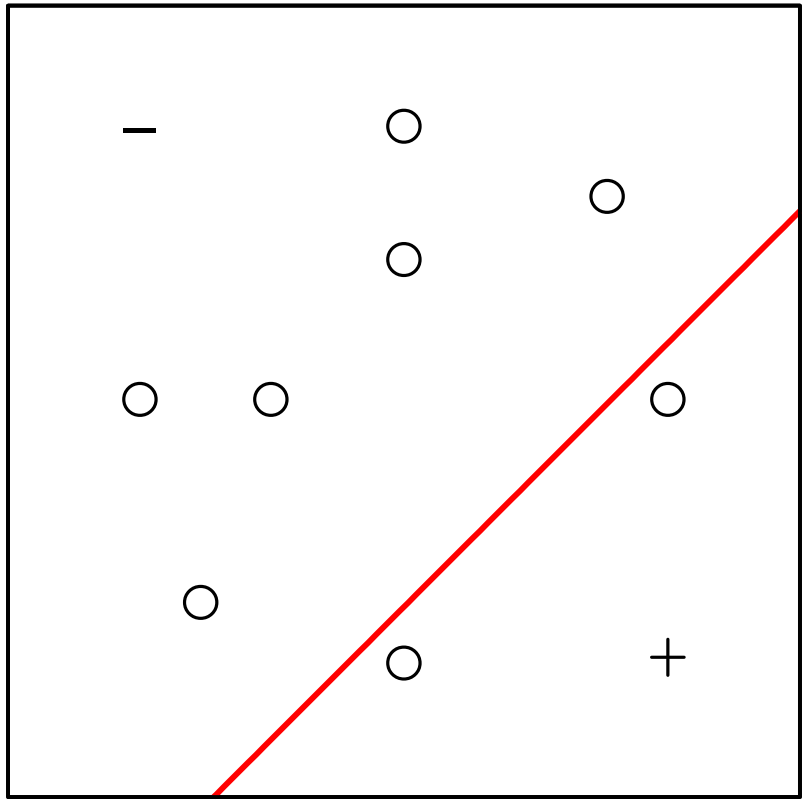
where $\hat{\mathbf{Y}}$ are "imagined" labels and $\phi[\hat{\mathbf{Y}}] = \text{count}(+, \hat{\mathbf{Y}})$



Visual Example: Posterior Regularization

$$\max_{\theta} \quad \log p_\theta(\mathbf{Y}_L|\mathbf{X}_L) - 0.1\|\theta\|_2^2 \; - \; \mathcal{D}_{\mathrm{KL}}(\mathcal{Q}||p_\theta)$$

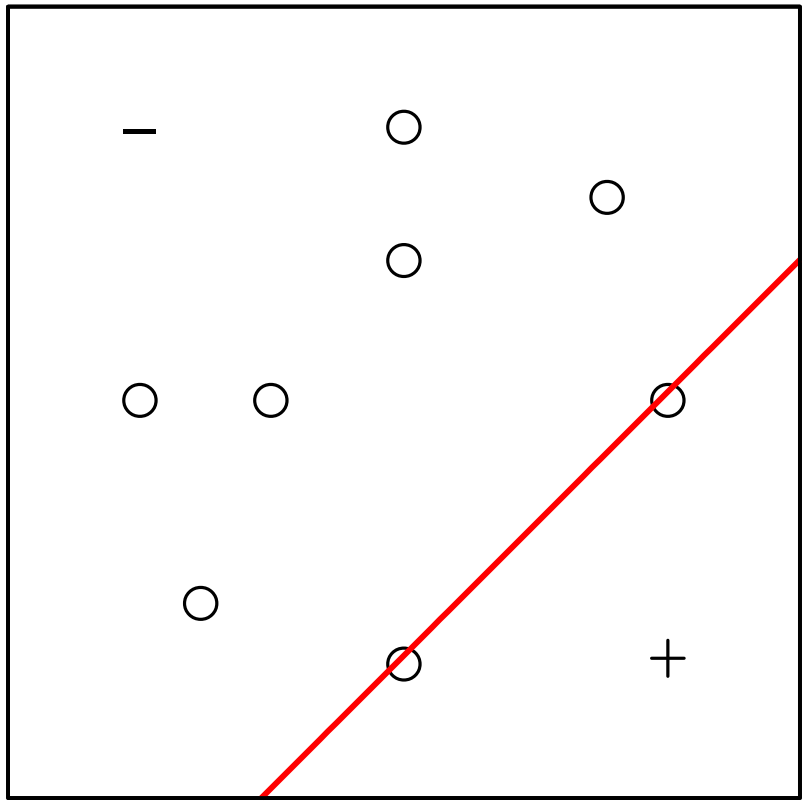
where: $\mathcal{D}_{\mathrm{KL}}(\mathcal{Q}||p_\theta) \; = \; \min_{q} \; \mathcal{D}_{\mathrm{KL}}(q||p_\theta) \;\; \text{s.t. } \mathbf{E}_q[\phi] = 2$



A visual comparison of the frameworks

Objective: Generalized Expectation Constraints

$$\max_{\theta} \quad \log p_\theta(\mathbf{Y}_L|\mathbf{X}_L) - 0.1\|\theta\|_2^2 - 500\|\mathbf{E}_{p_\theta}[\phi] - 2\|_2^2$$



API for New PR Constraints: MALLET

http://mallet.cs.umass.edu

- *Java Interfaces* for implementing **new** PR constraints
- inference algorithms implemented (MaxEnt, CRF)
- **primarily need to write code to:**
  - *compute constraint features*
  - *compute penalty and penalty-specific part of the gradient for the modified E-step*
- **restriction:** constraints must factor with model

Off-the-Shelf Tools & API: PR Toolkit

http://code.google.com/p/pr-toolkit/

- off-the-shelf support for **PR**
- **models:**
  - MaxEnt Classifier, HMM, DMV
- **applications:**
  - Word Alignment, Pos Induction, Grammar Induction
- **constraints:** posterior sparsity, bijectivity, agreement
- No command line mode
- Smaller support base

References

Many slides are from Gregory Druck, Kuzman Ganchev, João Graça. Rich prior knowledge in learning for NLP.

(1) Ganchev K, Graça J, Gillenwater J, et al. Posterior regularization for structured latent variable models[J]. The Journal of Machine Learning Research, 2010, 11: 2001-2049.

(2) Joao V. Graca, Lf Inesc-id, Kuzman Ganchev, Ben Taskar, Joo V. Graa, L F Inesc-id, Kuzman Ganchev, and Ben Taskar. 2007. Expectation maximization and posterior constraints. In In Advances in NIPS, pages 569–576.

(3) Chang M W, Ratinov L, Roth D. Guiding semi-supervision with constraint-driven learning[C]//Annual Meeting-Association for Computational Linguistics. 2007, 45(1): 280.

(4) Mann G S, McCallum A. Generalized expectation criteria for semi-supervised learning of conditional random fields[J]. 2008.

(5) Mann G S, McCallum A. Generalized expectation criteria for semi-supervised learning with weakly labeled data[J]. The Journal of Machine Learning Research, 2010, 11: 955-984.

(6) Mann G S, McCallum A. Simple, robust, scalable semi-supervised learning via expectation regularization[C]//Proceedings of the 24th international conference on Machine learning. ACM, 2007: 593-600.

(7) Li Zhao, Minlie Huang, Haiqiang Chen, Junjun Cheng, Xiaoyan Zhu. Clustering Aspect-related Phrases by Leveraging Sentiment Distribution Consistency. EMNLP 2014, October 25–29, 2014 — Doha, Qatar.

(8) Li zhao, Minlie Huang, Xiaoyan Zhu. Sentiment Extraction by Leveraging Aspect-Opinion Association Structure. CIKM 2015, Oct 19-23, Melbourne, Australia.

(9) Li Zhao, Minlie Huang, Ziyu Yao, Xiaoyan Zhu. Semi-supervised Multinomial Naive Bayes for Text Classification by Leveraging Word-level Statistical Constraint, accepted by AAAI'2016.

(10) Yang B, Cardie C. Context-aware Learning for Sentence-level Sentiment Analysis with Posterior Regularization[C]//ACL (1). 2014: 325-335.