Modeling Complete Natural Language Utterances Using MCMC Methods and Logistic Regression

Roni Rosenfeld

Carnegie Mellon University
Pittsburgh, Pennsylvania 15213
USA

Joint work with Stan Chen, Xiaojin Zhu, Larry Wasserman and Can Cai.

Outline

- INTRO
- What's wrong with today's language models?
- Whole sentence exponential language models
- Sampling and smoothing of exponential distributions
- Capturing syntactic features
- Shannon experiments
- An interactive methodology for linguistic feature induction
- Modeling semantic coherence
- Logistic regression for efficient training and feature construction

SEE JOURNAL PAPER

Existing Language Model Types

1. N-gram: $\Pr(w_i|w_1,\ldots,w_{i-1}) \approx \Pr(w_i|w_{i-N+1},\ldots,w_{i-1})$
   Issues: smoothing, clustering, shrinking.

2. Decision tree (using CART).
   Issues: finding a good tree.

3. Probabilistic Context Free Grammar (using EM).
   Issues: finding a good one, if it exists.

4. Exponential (Maximum Entropy) model (using GIS):

$$P(w|h) = \frac{1}{Z(h)} \cdot P_0(w|h) \cdot \exp[\sum_i \lambda_i f_i(h,w)]$$

   Best model so far.
   Issues: feature induction, computation (esp. normalization).

E.O. INTRO:

- NGRAM RULES IN PRACTICE
- BIGGEST PROBLEM: "PUTTING L BACK INTO LM" (JELINEK)
  - IN A SOUND STATISTICAL FRAMEWORK

The World of Broadcast News

- WANDILE ZOTHE DO YOU PERSONALLY KNOW PEOPLE WHO WERE ARRESTED AND TORTURED DURING THE APARTHEID ERA </s>
- SO HE PROBABLY WILL HAVE TO HAVE THEM TAXED BECAUSE THEY'RE NOT A TRADITIONAL PENSION FUND </s>
- BUT THE TOBACCO COMPANIES AND NASCAR OFFICIALS SAY THEIR FANS ARE WILDLY LOYAL TO RACE ADVERTISERS </s>
- THERE ARE A LOT OF QUALITY SWEATERS IN THE MARKET RIGHT NOW CASHMERE AND CASHMERE BLENDS </s>
- POLICE SAY THE MAN RAN FROM THE FRONT OF THE HOUSE AND CAME AROUND THIS CORNER </s>

The World According to Trigram

- THEY PUNISHED US A GROUP CALLED THE NEXT THING WE CAN COOPERATE TO DIFFUSE THIS TAPE </s>
- SAYING THAT HE DIDN'T WANT TO BE VERY GOOD ACTUALLY AFTER THE GULF WAR A REQUEST </s>
- MY QUESTION TO YOU THOSE PICTURES MAY STILL NOT IN ROMANIA AND I LOOKED UP CLEAN </s>
- YOU WERE GOING TO TAKE THEIR CUE FROM ANCHORAGE LIFTED OFF EVERYTHING WILL WORK SITE VERDI </s>
- ARE YOU REFERRING TO IS EXTREMELY RISKY BECAUSE I'VE BEEN TESTED WHOSE ONLY WITH A MAIN </s>

- Violates all global aspects of language
- Easy for people to tell apart 'real' from 'pseudo' sentence
- Should be easy to fix?

General Framework for Incorporating Linguistic Structure

Language Modeling Basics Revisited:

$$\Pr(s) = \Pr(w_1, \ldots, w_n) = \prod_{i=1}^{n} \Pr(w_i | w_1, \ldots, w_{i-1})$$

But:

- Chain rule not conducive to whole-sentence KSs
- Grammatical info expressed w.r.t. whole sentence, not $\Pr(w|h)$
- Example: sentence length
- Example: parsability

A Whole Sentence Exponential Model

$$\Pr(s) \stackrel{\text{def}}{=} \frac{1}{Z} \cdot P_0(s) \cdot \exp(\sum_i \lambda_i \cdot f_i(s)) \tag{1}$$

- $P_0(s)$ is an arbitrary initial model (e.g.: uniform, trigram)
- $f_i(s)$'s are *arbitrary computable properties of s*
  (fill in your favorite linguistic theory!)
- Sentence s is viewed as *bag of features*
- Z is a **universal** normalizing constant
- Has been used successfully for *conditional* modeling:

$$\Pr(w|h) \stackrel{\text{def}}{=} \frac{1}{Z(h)} \cdot P_0(w|h) \cdot e^{\sum_i \lambda_i \cdot f_i(h,w)}$$

Whole Sentence Maximum Entropy Training

Initialize:

1. Select features $f_i(s)$ (syntax, semantics, speech acts, ...)
2. Collect target expectations of features in a training set: $E_{\tilde{p}}[f_i]$
3. *Throw away the training set.*
4. Set initial parameters λ_i, thus defining a tentative $P_\lambda(s)$

Iterate until convergence:

1. Compute feature expectations under the current $P_\lambda(s)$:

$$E_{P_\lambda}[f_i] \stackrel{\text{def}}{=} \sum_s P_\lambda(s) f_i(s) \qquad \forall i$$

2. Compare to target expectations, and update the parameters:

$$\lambda_i \leftarrow \lambda_i + \log \frac{E_{\tilde{p}}[f_i]}{E_{P_\lambda}[f_i]}$$

Issues in ME/MDI Training:

1. Smoothing — prior on λ_i
2. Convergence rate — Step size F_i
3. Computational efficiency
4. Feature selection

Problems specific to whole-sentence models:

1. Cannot feasibly compute feature expectations

   $\Longrightarrow$ sample! $\quad E_P[f_i] \stackrel{\text{def}}{=} \sum_s P(s) f_i(s) \approx \frac{1}{M} \sum_{\substack{\text{sample} \\ \text{from } P}} f_i(s)$

   - Efficient sampling is the name of the game
   - Computational bottleneck: rare features, exact modeling

2. $P(s)$ is unnormalizable $\Longrightarrow$ normalization not needed!

Sampling from an Exponential Distribution

- Need to sample sentences from $P_{exp}(s) = \frac{1}{Z} P_0(s) e^{\sum_i \lambda_i \cdot f_i(s)}$
- No known efficient method

1. Importance Sampling
2. Monte Carlo Markov Chain (MCMC) methods:
   - Metropolis Sampling
   - Gibbs Sampling
   - Independence Sampling (hybrid Metropolis/Importance)

   weakness: efficiency hampered by correlations between successive sentences

Importance Sampling

- Idea: instead of sampling from P_{exp}, sample from some other distribution $P_{gen}(s)$ that is easier to generate from, then weigh the samples by $\frac{P_{exp}(s)}{P_{gen}(s)}$

$$E_{P_{exp}}[f_i] \stackrel{\text{def}}{=} \sum_s P_{exp}(s) f_i(s) = \sum_s P_{gen}(s) \frac{P_{exp}(s)}{P_{gen}(s)} f_i(S) = E_{P_{gen}}[\frac{P_{exp}}{P_{gen}} f_i]$$

- weakness: variance too great if $\frac{P_{exp}(s)}{P_{gen}(s)}$ or $\frac{P_{gen}(s)}{P_{exp}(s)}$ are large
- efficiency depends on distance between P_{exp} and P_{gen}
- we take $P_{gen}(s) = P_0(s)$ (the trigram-based distribution)
- for unnormalized distributions

$$E_{P_{exp}}[f_i] \approx \frac{\sum_{j=1}^{M} \frac{P_{exp}(s_j)}{P_{gen}(s_j)} f_i(s_j)}{\sum_{j=1}^{M} \frac{P_{exp}(s_j)}{P_{gen}(s_j)}}$$

Applying MCMC methods to Natural Language

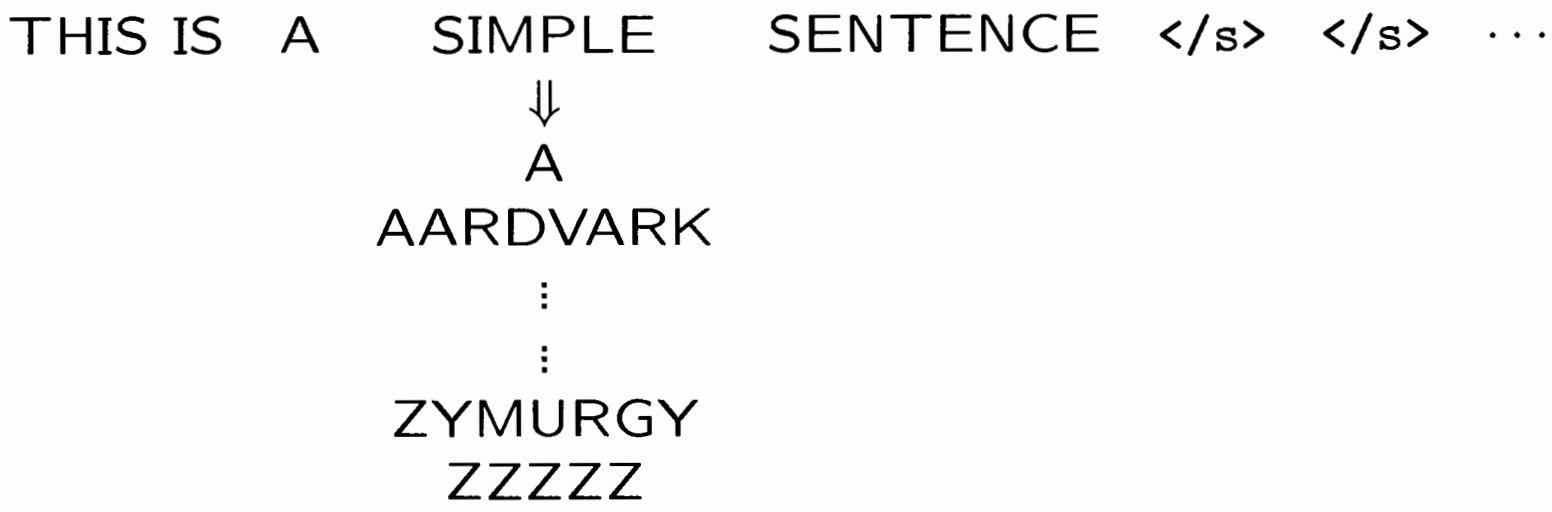
Language is:

- categorical (sort of)
- very high dimensional (typically 100,000)
- variable length
- neighbourhood? what neighbourhood?

$\Longrightarrow$ *as different from an Ising model as can be!*

- only requirement: "detail balance"

Gibbs Sampling of Utterances



- Sweep through the current utterance, one word at a time
- Replace the current word with a new word w, randomly selected according to the posterior $P_{exp}(w|\text{rest of utterance})$
- Requires efficient computation of the posterior
- Can be applied to word sequences of any length

Metropolis Sampling of Utterances

THIS IS A SIMPLE SENTENCE </s> </s> ...

$\Downarrow$

(Prob = P_{gen}) *propose* DIFFERENT

$\Downarrow$ *accept* (Prob = $\min\{1, \frac{P_{\text{exp}}(s_{\text{new}})P_{\text{gen}}(s_{\text{old}})}{P_{\text{exp}}(s_{\text{old}})P_{\text{gen}}(s_{\text{new}})}\}$)

THIS IS A DIFFERENT SENTENCE </s> </s> ...

$\Downarrow$

propose WHY

$\Downarrow$ *reject*

THIS IS A DIFFERENT SENTENCE </s> </s> ...

$\Downarrow$

propose THE

$\Downarrow$ *accept*

THIS IS A DIFFERENT SENTENCE THE </s> ...

- Distance between P_{exp} and P_{gen} determines efficiency
- Can be applied to word sequences of any length
- When applied to entire utterance $\longrightarrow$ Independence Sampling

Sampling Efficiency

sampling efficiency

- estimating fraction of sentences of certain lengths

	sampling algorithm		
	Metropolis	independence	importance
$f_{1,4}$	0.38±0.07	0.438±0.001	0.439±0.001
$f_{5,8}$	0.10±0.02	0.1001±0.0004	0.1001±0.0006
$f_{9,12}$	0.08±0.01	0.0834±0.0006	0.0831±0.0006
$f_{13,16}$	0.073±0.008	0.0672±0.0005	0.0676±0.0007
$f_{17,\infty}$	0.37±0.09	0.311±0.001	0.310±0.002

ME/MDI Smoothing

ME/MDI training is ML training

- $E_{\tilde{p}}[f_i] = 0 \quad \Rightarrow \quad$ for all $s : f_i(s) \neq 0, \;\; P(s) \rightarrow 0$

smoothing

- Gaussian prior on λ_i (variance σ^2)
- MAP training

$$E_P[f_i] = E_{\tilde{p}}[f_i] \qquad \Rightarrow \qquad E_P[f_i] = E_{\tilde{p}}[f_i] - \frac{\lambda_i}{\sigma^2}$$

- (modified) iterative scaling still guaranteed to converge

Experiment 1: Extended N-grams

$$P(s) = \frac{1}{Z} \cdot P_0(s) \cdot \exp\left(\sum_i \lambda_i f_i(s)\right)$$

Domain: SWITCHBOARD

Prior: $P_0(s)$ = probability of s acc. to (Kneser-Ney) trigram model

Features types:

- word n-grams (up to $n = 4$)
- class n-grams (up to $n = 5$, 1000 classes)
- distance-2 n-grams (up to $n = 3$)

$$f_\alpha(s) = \#\text{ of times } n\text{-gram } \alpha \text{ occurs in } s$$

Extended N-grams Feature Selection

find n-grams whose

- frequency in training data differs significantly from frequency according to prior trigram model
- by comparing counts of each n-gram in training corpus and in corpus generated with trigram model

n-gram	training	trigram
WHEN I GOING	1	0
WHEN I GOT	92	89
WHEN I GO	71	61
WHEN I WOULD	7	9
WHEN I CAN	18	14
WHEN I MEAN	2	23
WHEN I GOOD	0	1
WHEN I NOW	1	0
WHEN I I'VE	1	4
WHEN I KIND	0	3

Extended N-grams Feature Selection (cont.)

feature	training corpus count	trigram corpus count	χ^2
TALKING TO YOU KNOW	0	148	43512.50
TALKING TO _ KNOW	0	148	43512.50
TALKING/CHATTING TO YOU KNOW	0	148	43512.50
NICE/HUMONGOUS TALKING/CHATTING TO YOU KNOW	0	60	7080.50
HOW ABOUT YOU KNOW	0	56	6160.50
HOW ABOUT _ KNOW	0	56	6160.50
<s> HAVE _ KNOW	0	42	3444.50
KIND OF A WHILE/SUDDEN	0	42	3444.50
VAGUELY/BLUNTLY	15389	22604	3382.69

Experiment 1: Recognition Results

N-best list rescoring

- 8,300 word Switchboard/Call Home test set
- 200-best lists generated by Janus system (CMU)
- unnormalized model — PP difficult to calculate exactly

		χ^2 threshhold		
	trigram	100	30	15
# features	0	3.5k	19k	52k
WER	36.53	36.49	36.37	36.29
LM only	40.92	40.95	40.68	40.46

Experiment 1: Recognition Results (cont.)

	trigram	ME/MDI $\chi^2 \geq 15$	*word 4-gram*	*word 4-gram + class 5-gram*
# features	0	52k	2.1M	7.9M
WER	36.53	36.29	36.21	35.95
LM only	40.92	40.46	40.52	40.03

performance by feature type ($\chi^2 \geq 15$)

	trigram	all	word n-grams	class n-grams	distance-2 n-grams
# features	0	52k	14k	20k	19k
WER	36.53	36.29	36.51	36.34	36.37
LM only	40.92	40.46	40.71	40.75	40.76

Experiment 1: Perplexity Estimation

- Normalization estimation

$$P(s) = \frac{1}{Z} \cdot P_0(s) \cdot \exp\left(\sum_i \lambda_i f_i(s)\right) \stackrel{\text{def}}{=} \frac{1}{Z} \cdot P^*(s)$$

$$Z = \frac{P^*(s)}{P(s)}$$

	trigram	ME/MDI $\chi^2 \geq 100$	*word 4-gram*	*word 4-gram + class 5-gram*
PP	81.4	80.6	80.5	77.6

Feature Induction

How can we take advantage of the whole-sentence paradigm to find useful features?

- Original corpus and trigram-generated ('fake') corpus
- Pose a challenge: find (computable) differences
- Course project for 3 students:

  1. unigram marginals (surprise!)
  2. distance-k class ngrams
  3. parse based features

Experiment 2: Parse Based Features

- Use Klaus Zechner's shallow Switchboard parser
- Parser maps sentence into variable-length constituents: NP, ADJ, VB, AUX, ...
- 3 feature types:
  - constituent strings, *e.g.* $f_{\text{np-vb-prep-adj-np}}(s)$
  - constituent sets, *e.g.* $f_{\text{np-vb-prep-adj}}(s)$
  - constituent trigram, *e.g.* $f_{\text{np-vb-prep}}(s)$, $f_{\text{vb-prep-adj}}(s)$, ...
- Found 7,000 features with significant (T, T_0) discrepancy
- Example: $f_{\text{conj-np-aux-adj}}(s)$
  (Never occurred in the original SWB corpus, but occurred 19 times in the trigram-generated corpus, e.g. "and you can convenient".)
- $\Longrightarrow$ slight improvement in PP & recognition

Experiment 2: Results

- Perplexity: 81.37 $\Longrightarrow$ 80.49
- N-best rescoring WER: 36.53% $\Longrightarrow$ 36.38%

Analysis

Q: Why aren't we making a bigger difference?

- Upper bound on improvement from feature f_i is the K-L distance between $\tilde{P}(f_i)$ (the *true* distribution of f_i and $P(f_i)$ (the current model's distribution of f_i).
- For the parse-based features: $\sum_i D(\tilde{P}(f_i) \| P_0(f_i)) = 0.062$
- So perplexity can improve by at most ~~0.25~~ 43% $(2^{\frac{0.062}{\cancel{17}\ 10}})$

Spell out: $P \cdot \log \frac{P}{Q}$

A: need more powerful (i.e. common) features!

- $f_i(s)$ = "sentence s makes sense" ??

"Makes Sense" Feature: A Shannon Experiment

- 17 members of the Sphinx research group
- 40 sentences (20 "real", 20 trigram-generated

	human error
1. original sents	10% ± 5%
2. common words removed	34% ± 9%
3. +trigram neighborhood removed	38% ± 10%

Consider the features $f_{\text{makes sense}}(s)$ that people presumably relied on:

$D(\tilde{P}(f_1) \| P_0(f_1)) = 3.0 \quad \Longrightarrow$ 12% PP reduction
$D(\tilde{P}(f_2) \| P_0(f_2)) = 0.3 \quad \Longrightarrow$ 1.2% PP reduction
$D(\tilde{P}(f_2) \| P_0(f_2)) = 0.01 \quad \Longrightarrow$ 0.7% PP reduction

A Methodology for Feature Induction

Given corpus T of training sentences:

1. Train best-possible baseline model, $P_0(s)$
2. Use $P_0(s)$ to generate corpus T_0 of "pseudo sentences"
3. Pose a challenge: find (computable) differences
   - Does P_0 adequately model your favorite linguistic aspects?
4. Encode the differences as features $f_i(s)$
5. Train a new mode: $P_1(s) = \frac{1}{Z} \cdot P_0(s) \cdot e^{\sum_i \lambda_i \cdot f_i(s)}$
6. Use $P_1(s)$ to generate corpus T_1 of "pseudo sentences"
7. Go to step 3

- Emphasis is on the "human in the loop"

In Search of Computable Differences

- WANDILE ZOTHE DO YOU PERSONALLY KNOW PEOPLE WHO WERE ARRESTED AND TORTURED DURING THE APARTHEID ERA </s>
- SO HE PROBABLY WILL HAVE TO HAVE THEM TAXED BECAUSE THEY'RE NOT A TRADITIONAL PENSION FUND </s>
- BUT THE TOBACCO COMPANIES AND NASCAR OFFICIALS SAY THEIR FANS ARE WILDLY LOYAL TO RACE ADVERTISERS </s>
- THERE ARE A LOT OF QUALITY SWEATERS IN THE MARKET RIGHT NOW CASHMERE AND CASHMERE BLENDS </s>
- THEY PUNISHED US A GROUP CALLED THE NEXT THING WE CAN COOPERATE TO DIFFUSE THIS TAPE </s>
- SAYING THAT HE DIDN'T WANT TO BE VERY GOOD ACTUALLY AFTER THE GULF WAR A REQUEST </s>
- MY QUESTION TO YOU THOSE PICTURES MAY STILL NOT IN ROMANIA AND I LOOKED UP CLEAN </s>
- YOU WERE GOING TO TAKE THEIR CUE FROM ANCHORAGE LIFTED OFF EVERYTHING WILL WORK SITE VERDI </s>

Going After Semantic (in)Coherence

- Define "content words" (all but top 200; 40% of tokens)
- Model distribution of content words in sentence
- Simplify: model pairwise co-occurrences ("content word pairs")
- Exclude trigram effects
- Collect Contingency tables:

Semantic Association between Content WordPairs

- SO HE PROBABLY WILL HAVE TO HAVE THEM TAXED BECAUSE THEY'RE NOT A TRADITIONAL PENSION FUND </s>

- Contingency table:

TAXED \ FUND	Yes	No
Yes	C_{11}	C_{12}
No	C_{21}	C_{22}

C_{11} =Number of sentences containing content wordpair FUND TAXED

$C_{12} = C_2 - C_{11}$

$C_{21} = C_1 - C_{11}$

$C_{22} = N - C_{11} - C_{12} - C_{21}$

Measures of Association

- Correlation coefficient:

$$\hat{\rho} = \frac{C_{11}C_{22} - C_{12}C_{21}}{\sqrt{C_{1+}C_{2+}C_{+1}C_{+2}}}$$

- Pearson chi-square
- Mutual information:

$$\hat{I} = \sum_{i,j=1,2} \frac{C_{ij}}{N} \log N \left(\frac{C_{ij}}{C_{i+}C_{j+}} \right)$$

- Yule's measure of association:

$$\hat{Q} = \frac{C_{11}C_{22} - C_{12}C_{21}}{C_{11}C_{22} + C_{12}C_{21}}$$

Examples Q values

ENTERTAINMENT WITNESS -0.89

ANGELES CONGRESS -0.70

ABORTION REPUBLICAN 0.74

ARTISTS SONY 0.89

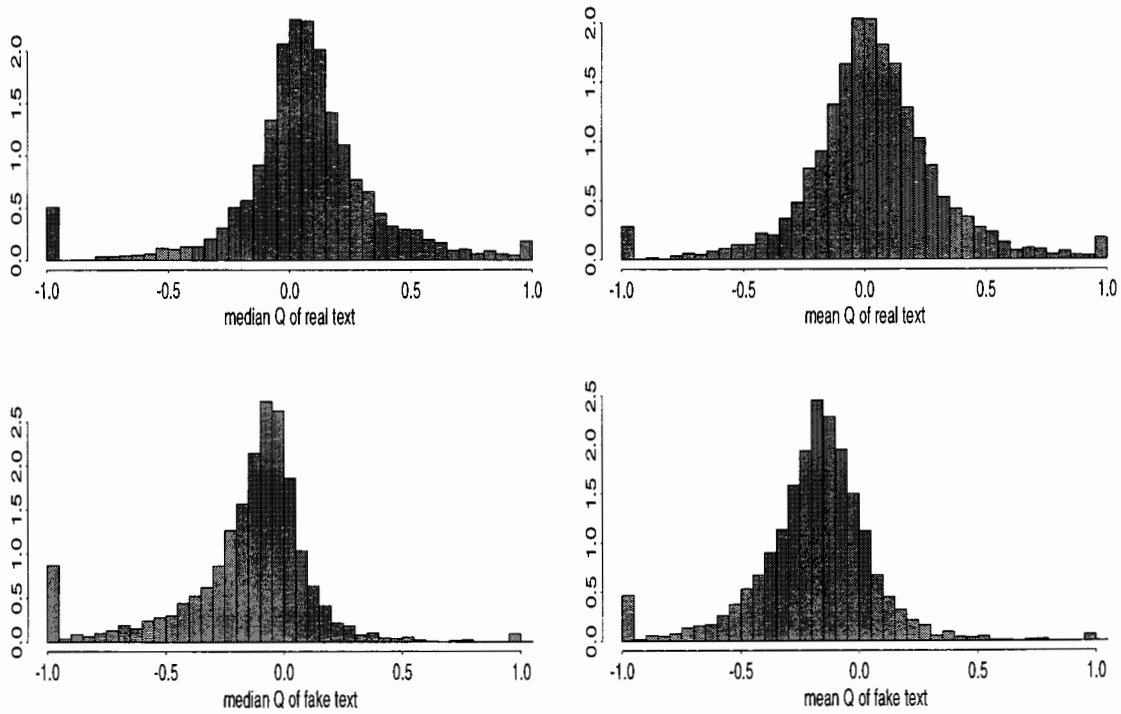
M H 1 (think M * A * S * H)

Modeling Semantic Coeherence

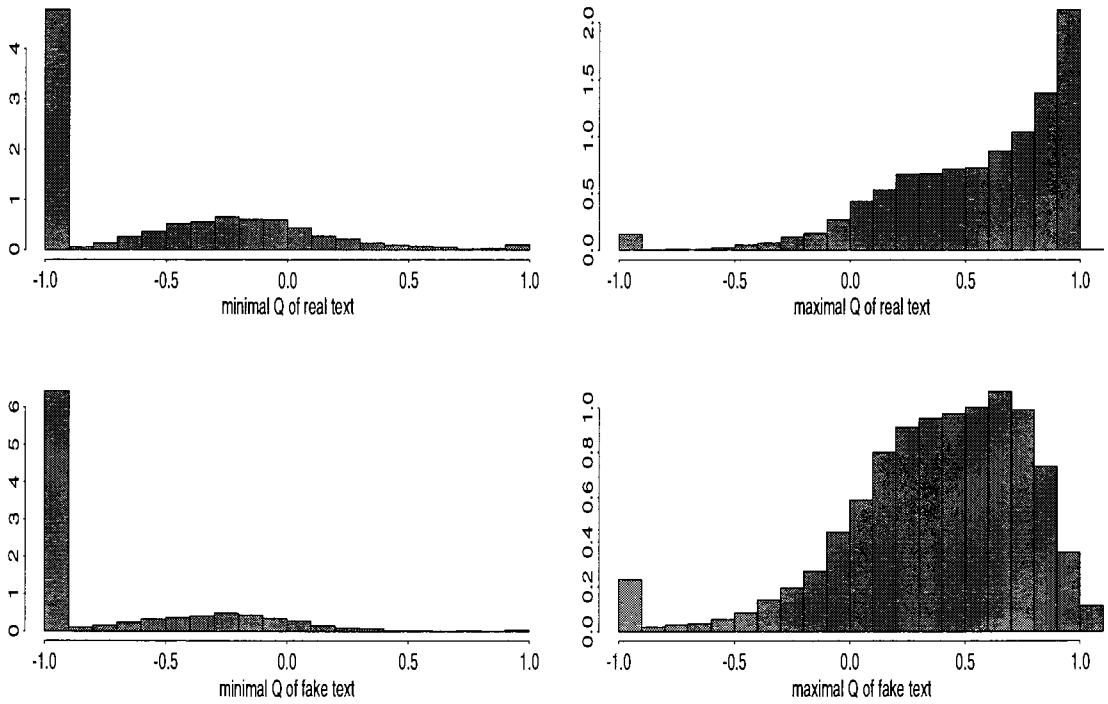
- Collect contingency table counts, and Q statistics, for all content wordpairs in the training data.
- For each sentence in the 'true' test set and the 'pseudo' (trigram -generated) set:

$$\text{A sentence}$$
$$\Downarrow$$
$$W_1W_2,\ W_1W_3, \cdots$$
$$\Downarrow$$
$$Q_{12}, Q_{13}, \cdots$$
$$\Downarrow$$
$$Q_{\text{mean}}, Q_{\text{median}}, Q_{\text{max}}, Q_{\text{min}}$$

Comparing Q Statistics



Comparing Q Statistics (cont.)



From ML Estimation to Logistic Regression

- How to best exploit these distributional differences?
- Use the Q stats as features
- Recall: $P(s;\lambda) = \frac{1}{Z} P_0(s) \cdot \exp(\sum_i \lambda_i f_i(s))$
- Find the MLE
- Alternatively, convert into a discrimination problem, and use regression

Exponential Model Fitting by Logistic Regression

- Let $s_1, \ldots s_n$ be "real" sents (supposedly drawn from $P(s)$)
- Let $s_{n+1}, \ldots, s_{2n}$ be "pseudo" sents, gen'ed by baseline $P_0(s)$
- Seek function $h(s)$ that maximally discriminates P from P_0
- Let $Y = \begin{cases} 1 & s \in P \\ 0 & s \in P_0 \end{cases}$
- Let $h(s) = P(Y = 1|s)$

$$\begin{aligned} h(s) &= P(Y = 1|s) \\ &= \frac{P(s|Y = 1)P(Y = 1)}{P(s|Y = 1)P(Y = 1) + P(s|Y = 0)P(Y = 0)} \\ &= \frac{P(s)}{P(s) + P_0(s)} \end{aligned}$$

- $\frac{h(s)}{1-h(s)} = \frac{P(s)}{P_0(s)}$

Logistic Regression (cont.)

- $\frac{h(s)}{1-h(s)} = \frac{P(s)}{P_0(s)}$
- Recall $P(s;\lambda) = \frac{1}{Z} P_0(s) \cdot \exp(\sum_i \lambda_i f_i(s))$

$$\begin{aligned} \log[\frac{h(s)}{1-h(s)}] &= -\log Z + \sum_i \lambda_i f_i(s) \\ &= \beta_0 + \sum_i \beta_i f_i(s) \end{aligned}$$

- For logistic regression:

$$\text{logit}(s) = \beta_0 + \sum_i \beta_i f_i$$

- For generalized additive model (GAM):

$$\text{logit}(s) = s_0 + \sum_i s(f_i)$$

Why Logistic Regression?

- Fitting logistic regression is *much* easier than Iterative Scaling (no sampling, no normalization)
- Tap into huge existing base of methods and software
- Can do "model selection" (ie play with features) very easily
- Like MLE, estimate is unbiased, asymptotically normal convergence
- Estimation is not as data-efficient as MLE (loss of information given by the Fisher Information). But there's unlimited P_0 data!
- Generalized Additive models: efficiently search for non-linear combinations of the given features

Results from Regression

- Logistic regression:

$$\log[\frac{h(s)}{1-h(s)}] = \beta_0 + \beta_1 f_1(s) + \ldots + \beta_6 f_6(s)$$

	Coefficient	t value
(Intercept)	0.02	1.83
Q_min	-1.30	-32.40
Q_med	-1.81	-25.99
Q_max	-0.12	-3.00
Q_mean	6.20	51.69
# words	-0.007	-6.55
# wordpairs	-0.002	-9.79

Results from GAM

$$\log[\frac{h(s)}{1-h(s)}] = \beta_0 + s_1(f_1(s)) + \ldots + s_6(f_6(s))$$

	Chisq	P(Chi)
s(Q_min)	533.63	0
s(Q_med)	1308.84	0
s(Q_max)	6185.01	0
s(Q_mean)	315.21	0
s(#words)	144.01	0
s(#wordpairs)	496.24	0

What About Perplexity?

- $P(s;\lambda) = \frac{1}{Z} P_0(s) \cdot \exp(\sum_i \lambda_i f_i(s))$
- Features modify the probability of the *entire* sentence
- Effect of single feature on *per-word* probability is very small
- Using only the 6 features above, PP reduction was:
  1. Linear logistic regression: 29% per sentence (1.5% per word)
  2. GAM model: 79% per sentence (3.5% per word)

Exponential Models: ML vs. MCE

- ME/MDI is Maximum Likelihood Estimation within the exponential family
- Can train exponential model to directly reduce WER:

$$P(s) = \frac{1}{Z} \cdot P_0(s) \cdot e^{\sum_i \lambda_i \cdot f_i(s)}$$

$$\log P(s) = \text{Const.} + \log P_0(s) + \sum \mu_i f_i(s)$$

- Minimize WER by heuristically searching over the μ_i's (Powell's algorithm)

Summary

- Whole-Sentence ME: A framework for modeling language
  - Facilitates long-range and whole-sentence modeling
  - Opens the door to "Putting language back into 'Language Modeling"' (Jelinek, 1995)
  - Focuses attention on feature induction
- Methodology for feature induction
  - Put a human in the loop
  - Model is still optimized on data
  - Integrates linguistic intuition with statistical methodology
- Modeling Semantic Coherence
  - Major weakness of current models
  - For now, global stats of pairwise content word correlations
- Logistic regression
  - Parametric regression for efficient training
  - Non-parametric regression for powerful feature construction

Linguistic Structure in Statistical Language Models

Why aren't we there yet?

1. Linguistic theories deal with existence, SLM with prevalence

2. Lack of general framework
   - each linguistic theory poses a new estimation problem

3. Mental straightjacket of the conditional formulation ($P(w|h)$)

4. Non-informative priors (Bayesian view)
   - Data will always be sparse — good prior is crucial
   - Must encode linguistic knowledge as prior

   - We still don't know how to do that!

The Catch-22 of Non-Informative Priors

Example: Vacabulary Clustering

- Many greedy algorithms [IBM, Philips, . . .]
- Can be seeded with POS information
- Example classes from [Chen, unpublished]:
  - MY THY JESSICA'S SARAH'S KEVIN'S CONGESTIVE KAREN'S HEIDI'S
  - THEN THEREFORE CONSEQUENTLY THIRDLY LASTLY BEHOLD FRO ABETTIN
  - DOWN ASIDE ASHORE INS OVERBOARD IDLY ... AFIRE ROUGHSHOD
  - LET EXCUSE FORGIVE PARDON TICKLE
  - STATE CENSUS COMMONWEALTH PROVISIONAL FOOTHILLS
  - WASHINGTON LONDON MOSCOW PARIS TOKYO ... ISLAMABAD EDGEWISE
- Catch: rarest words clustered least well
- Mildly successful