Language Modeling for Automatic Speech Recognition Meets the Web:

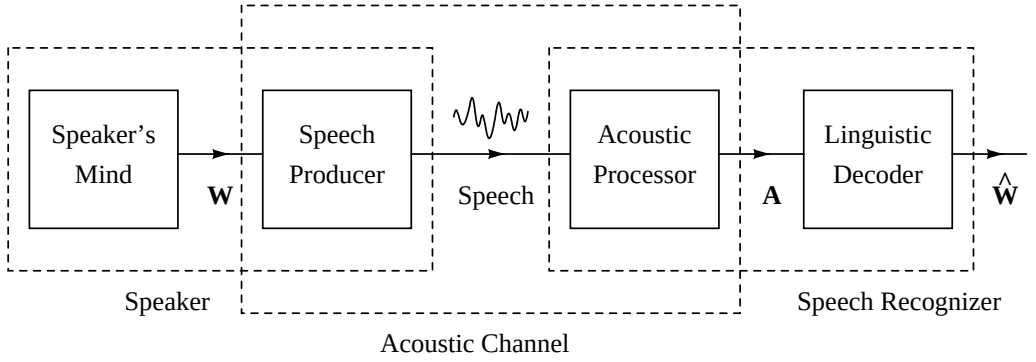
Google Search by Voice

Ciprian Chelba, Johan Schalkwyk, Boulos Harb, Carolina Parada, Cyril Allauzen,

Michael Riley, Peng Xu, Thorsten Brants, Vida Ha, Will Neveitt

Google

Statistical Modeling in Automatic Speech Recognition



$$\hat{W} = \text{argmax}_{\text{W}} P(W|A) = \text{argmax}_{\text{W}} P(A|W) \cdot P(W)$$

- $P(A|W)$ *acoustic model* (Hidden Markov Model)
- $P(W)$ *language model* (Markov chain)
- *search* for the most likely word string $\hat{W}$
  - due to the large vocabulary size—1M words—an exhaustive search is intractable

Google

Language Model Evaluation (1)

Word Error Rate (WER)

```
TRN: UP UPSTATE NEW YORK SOMEWHERE UH      OVER
HYP:    UPSTATE NEW YORK SOMEWHERE UH ALL  ALL
     D  0       0   0    0         0  I    S
   :3 errors/7 words in transcript; WER = 43%
```

Perplexity(PPL)

$$PPL(M) = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\ln\left[P_M(w_i|w_1\ldots w_{i-1})\right]\right)$$

- good models are smooth: $P_M(w_i|w_1\ldots w_{i-1}) > \epsilon$
- other metrics: out-of-vocabulary rate/n-gram hit ratios

Google

Language Model Evaluation (2)

Web Score (WebScore)

```
TRN: TAI PAN RESTAURANT PALO ALTO
HYP: TAIPAN RESTAURANTS PALO ALTO
```

- produce the same search results
- do not count as error if top search result is identical with that for the manually transcribed query

Language Model Smoothing

Markov assumption:

$$P_\theta(w_i/w_1 \dots w_{i-1}), \theta \in \Theta, w_i \in \mathcal{V}$$

Smoothing using Deleted Interpolation:

$$\begin{aligned} P_n(w|h) &= \lambda(h) \cdot P_{n-1}(w|h') + (1 - \lambda(h)) \cdot f_n(w|h) \\ P_{-1}(w) &= uniform(\mathcal{V}) \end{aligned}$$

Parameters (smoothing weights $\lambda(h)$ must be estimated on cross-validation data):

$$\theta = \{\lambda(h); count(w|h), \forall (w|h) \in \mathcal{T}\}$$

Google

Voice Search LM Training Setup

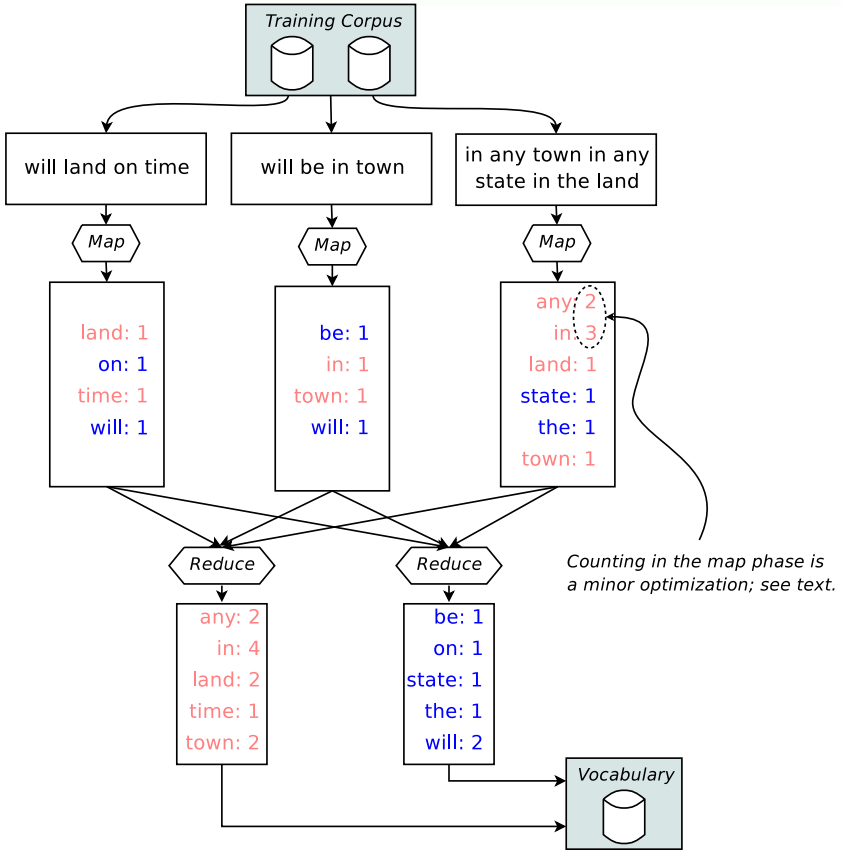
- correct[a] google.com queries, normalized for ASR, e.g. `5th -> fifth`
- vocabulary size: 1M words, OoV rate 0.57% (!), excellent n-gram hit ratios
- training data: 230B words

Order	no. n-grams	pruning	PPL	n-gram hit-ratios
3	15M	entropy	190	47/93/100
3	7.7B	none	132	97/99/100
5	12.7B	1-1-2-2-2	108	77/88/97/99/100

[a] Thanks Mark Paskin

Google

Distributed LM Training



- Input: key=ID, value=sentence/doc
- Intermediate: key=word, value=1
- Output: key=word, value=count
- Map chooses reduce shard based on hash value (red or bleu) [a]

[a] T. Brants et al., *Large Language Models in Machine Translation*

Using Distributed LMs

- load each shard into the memory of one machine
- Bottleneck: in-memory/network access at X-hundred nanoseconds/Y milliseconds (factor 10,000)

Example: translation of one sentence

- approx. 100k n-grams; 100k * 7ms = 700 seconds per sentence
- Solution: batched processing
- 25 batches, 4k n-grams each: less than 1 second [a]

[a] T. Brants et al., *Large Language Models in Machine Translation*

Google

ASR Decoding Interface

First pass LM: finite state machine (FSM) API

- states: n-gram contexts
- arcs: for each state/context, list each n-gram in the LM + back-off transition
- trouble: need all n-grams in RAM (tens of billions)

Second pass LM: lattice rescoring

- states: n-gram contexts, after expansion to rescoring LM order
- arcs: {new states} X {no. arcs in original lattice}
- good: distributed LM and large batch RPC

Google

Language Model Pruning

Entropy pruning is required for use in 1st pass:

- should one remove n-gram (h, w)?

$$D[q(h)p(\cdot|h) \parallel q(h) \cdot p'(\cdot|h)] = q(h) \sum_w p(w|h) \log \frac{p(w|h)}{p'(w|h)}$$

$$| D[q(h)p(\cdot|h) \parallel q(h) \cdot p'(\cdot|h)] | < pruning\ threshold$$

- lower order estimates: $q(h) = p(h_1) \dots p(h_n|h_1 \dots h_{n-1})$ or relative frequency: $q(h) = f(h)$
- very effective in reducing LM size at min cost in PPL

Google

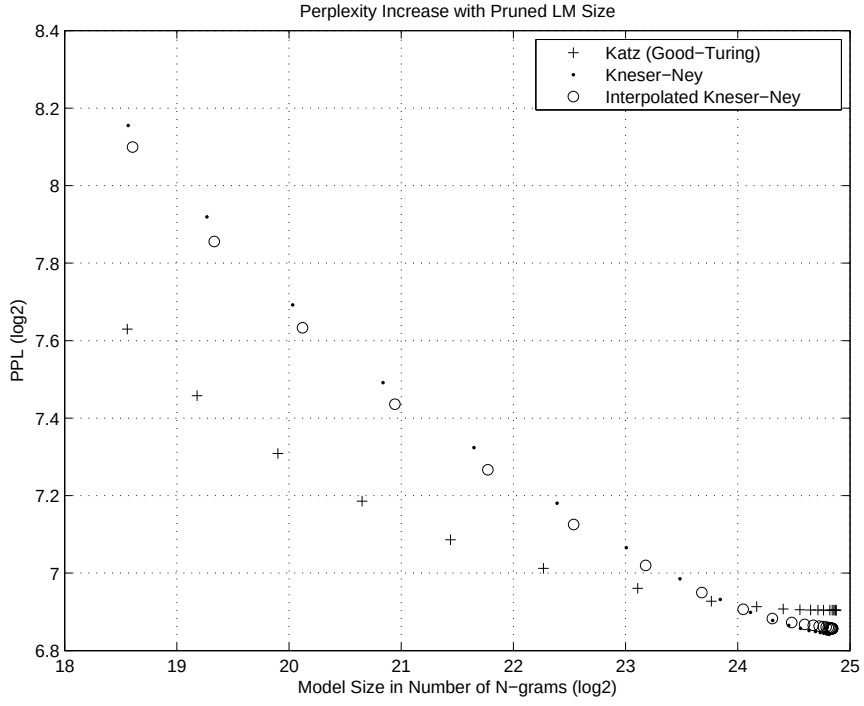
On Smoothing and Pruning (1)

- 4-gram model trained on 100Mwds, 100k vocabulary, pruned to 1% of raw size using SRILM
- tested on 690k wds

4-gram LM smoothing	**Perplexity** raw	pruned
Ney	120.5	197.3
Ney, Interpolated	119.8	198.1
Witten-Bell	118.8	196.3
Witten-Bell, Interpolated	121.6	202.3
Ristad	126.4	203.6
Katz (Good-Turing)	119.8	198.1
Kneser-Ney	114.5	285.1
Kneser-Ney, Interpolated	115.8	274.3
Kneser-Ney (CG)	116.3	280.6
Kneser-Ney (CG, Interpolated)	115.8	274.3

Google

On Smoothing and Pruning (2)



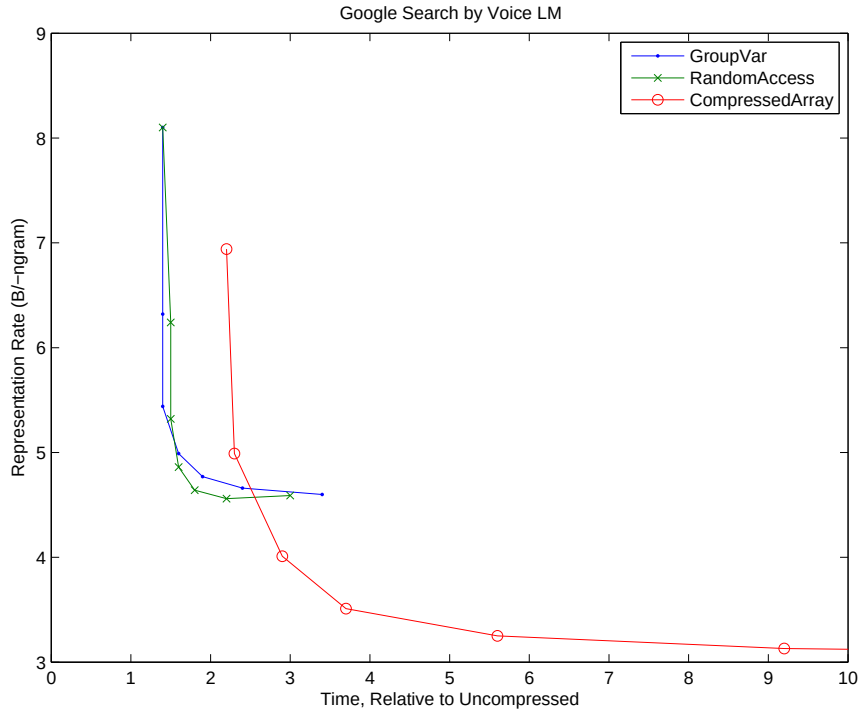
- baseline LM is pruned to 0.1% of raw size!
- switch from KN to Katz smoothing: 10% WER gain

Billion n-gram 1st Pass LM (1)

LM representation rate

Compression Technique	Block Length	Rel. Time	Rep. Rate (B/n-gram)
None	—	1.0	13.2
Quantized	—	1.0	8.1
CMU 24b, Quantized	—	1.0	5.8
GroupVar	8	1.4	6.3
	64	1.9	4.8
	256	3.4	4.6
RandomAccess	8	1.5	6.2
	64	1.8	4.6
	256	3.0	4.6
CompressedArray	8	2.3	5.0
	64	5.6	3.2
	256	16.4	3.1

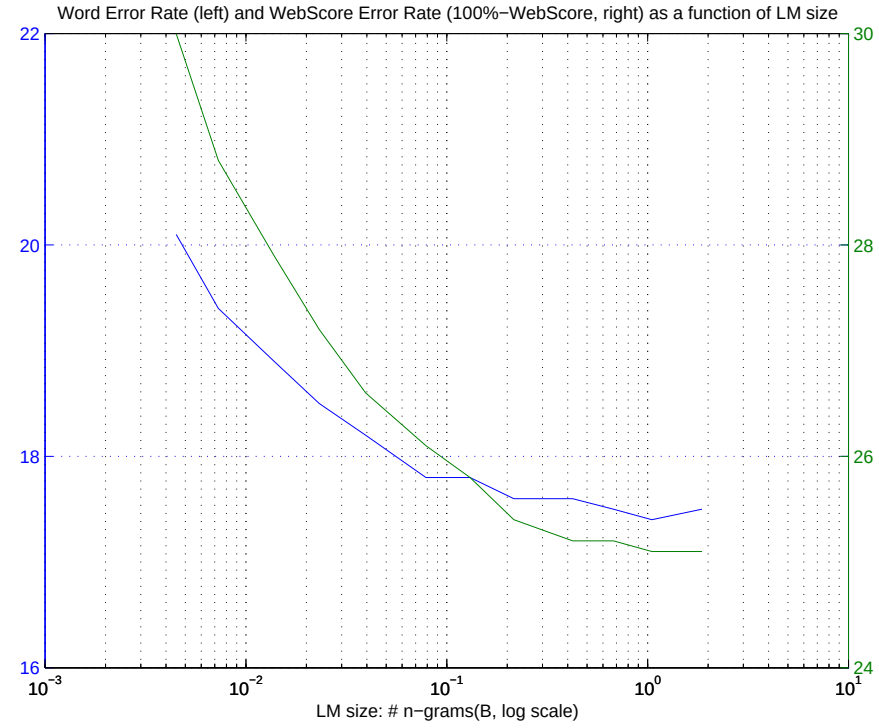
Billion n-gram 1st Pass LM (2)



- 1B 3-grams: 5GB of RAM @acceptable lookup speed[a]

[a] B. Harb, C. Chelba, J. Dean and S. Ghemawat, *Back-Off Language Model Compression, Interspeech 2009*

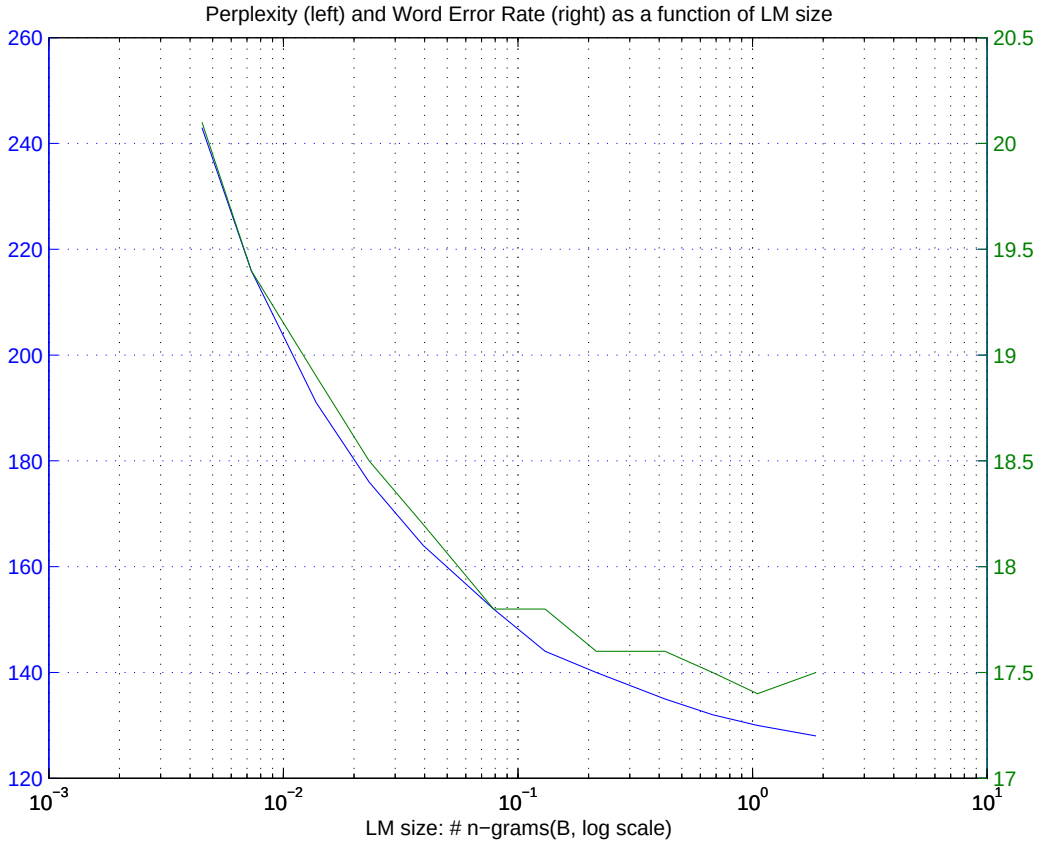
Is Bigger Better? YES!



- 8%/10% relative gain in WER/WebScore[a]

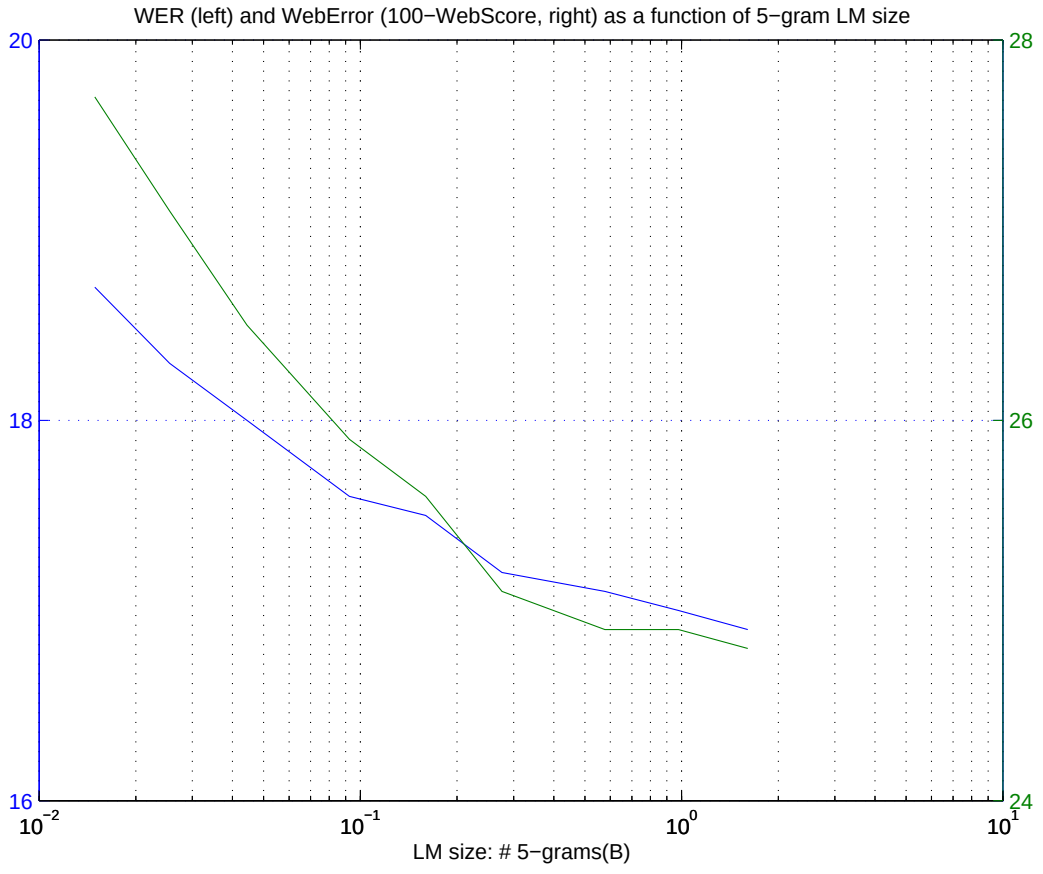
[a] With Cyril Allauzen, Johan Schalkwyk, Mike Riley, *May reachable composition CLoG be with you!*

Is Bigger Better? YES!



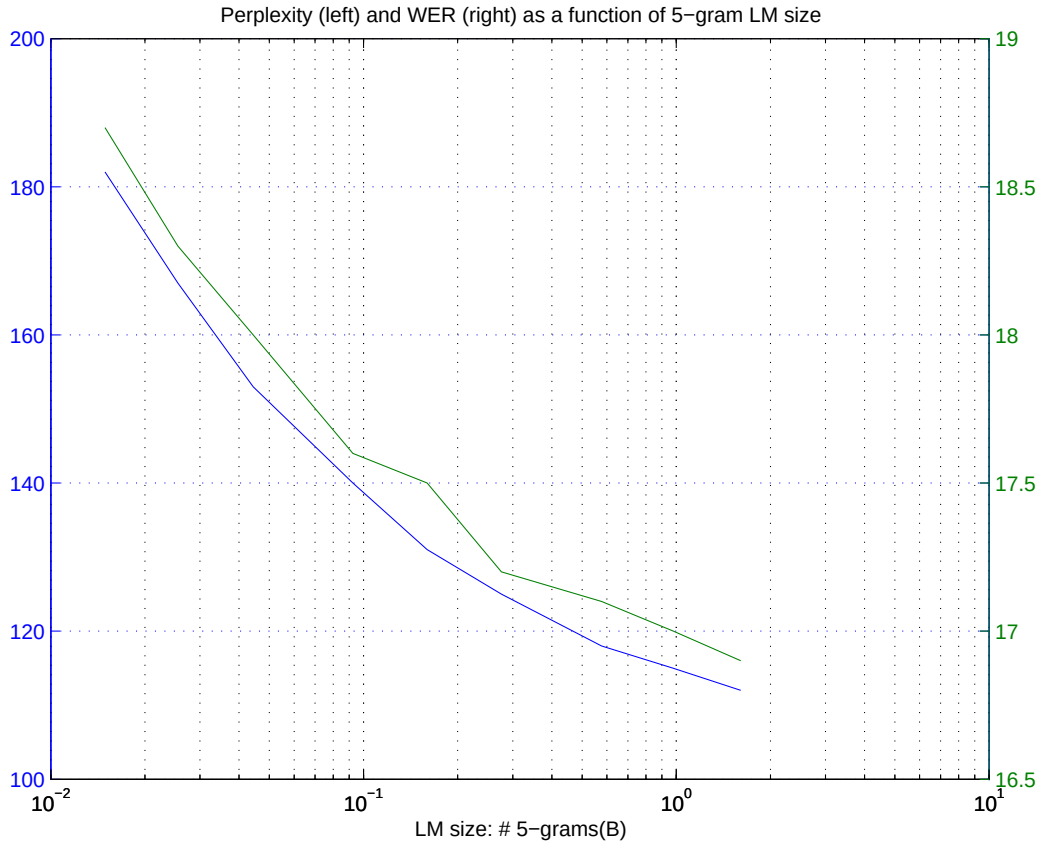
- PPL is really well correlated with WER!

Is Even Bigger Better? YES!



- 5-gram: 11% relative in WER/WebScore

Is Even Bigger Better? YES!



- Again, PPL is really well correlated with WER!

Detour: Search vs. Modeling error

$$\hat{W} = \text{argmax}_W P(A, W|\theta)$$

If correct $W^* \neq \hat{W}$ we have an error:

- $P(A, W^*|\theta) > P(A, \hat{W}|\theta)$: search error
- $P(A, W^*|\theta) < P(A, \hat{W}|\theta)$: modeling error
- wisdom has it that in ASR
  search error $<$ modeling error
- Corollary: improvements come primarily from using better models, integration in decoder/search is second order!

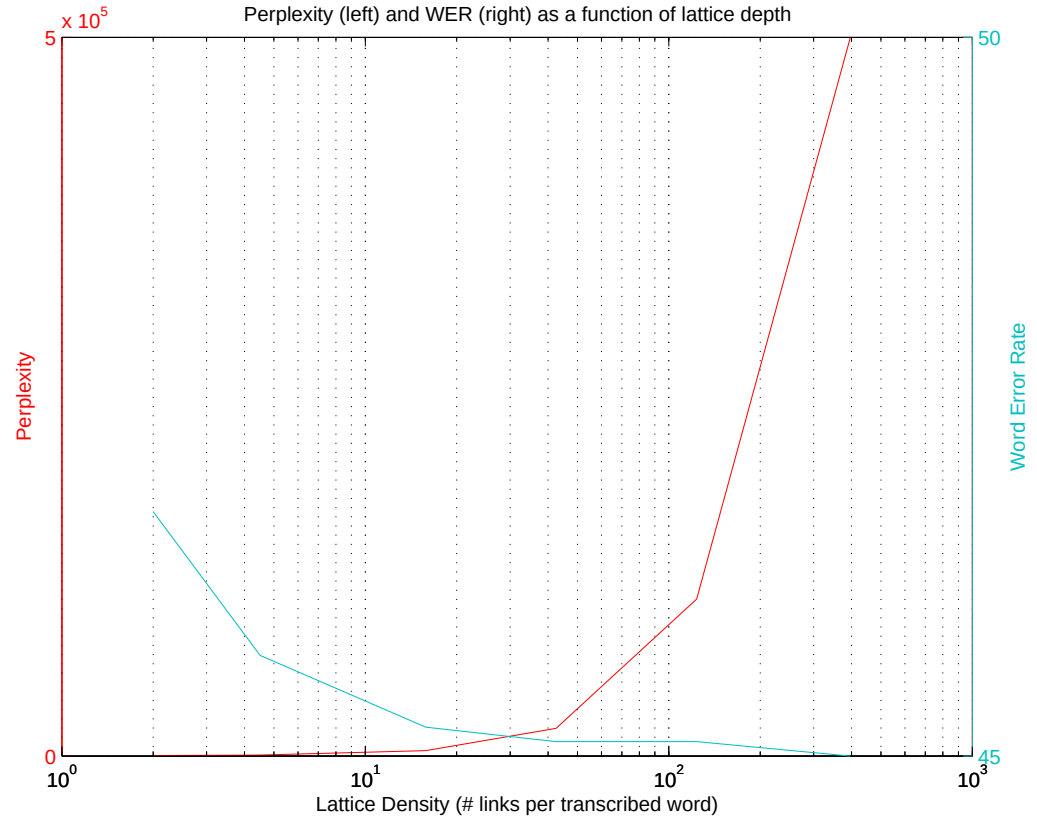
Lattice LM Rescoring

Pass	Language Model	PPL	WER	WebScore
1st	15M 3g	191	18.7	72.2
1st	1.6B 5g	112	16.9	75.2
2nd	15M 3g	191	18.8	72.6
2nd	1.6B 3g	112	16.9	75.3
2nd	12B 5g	108	16.8	75.4

- 10% relative reduction in remaining WER, WebScore error
- 1st pass gains matched in ProdLm lattice rescoring[a] at negligible impact in real-time factor

[a] Older front end, 0.2% WER diff

Lattice Depth Effect on LM Rescoring



- LM becomes ineffective after a certain lattice depth

N-best Rescoring

- N-best rescoring experimental setup
- minimal coding effort for testing LMs: all you need to do is assign a score to a sentence

Experiment	LM	WER	WebScore
SpokenLM baseline	13M 3g	17.5	73.3
lattice rescoring	12B 5g	16.1	76.3
10-best rescoring	1.6B 5g	16.4	75.2

- a good LM will immediately show its potential, even on as little as 10-best alternates rescoring!

Google

Query Stream Non-stationarity (1)

- USA training data[a]:
  - XX months
  - X months
- test data: 10k, Sept-Dec 2008[b]
- very little impact in OoV rate for 1M wds vocabulary: 0.77% (X months vocabulary) vs. 0.73% (XX months vocabulary)

---

[a] Thanks Mark Paskin

[b] Thanks Zhongli Ding for query selection.

Query Stream Non-stationarity (2)

3-gram LM	Training Set	Test Set PPL
unpruned	X months	121
unpruned	XX months	132
entropy pruned	X months	205
entropy pruned	XX months	209

- bigger is not always better[a]
- 10% rel reduction in PPL when using the most recent X months instead of XX months
- no significant difference after pruning, in either PPL or WER

[a] The vocabularies are mismatched, so the PPL comparison is a bit troublesome. The difference would be higher if we used a fixed vocabulary.

More Locales

- training data across 3 locales[a]: USA, GBR, AUS, spanning same amount of time ending in Aug 2008
- test data: 10k/locale, Sept-Dec 2008

Out of Vocabulary Rate:

Training Locale	Test Locale USA	GBR	AUS
USA	0.7	1.3	1.6
GBR	1.3	0.7	1.3
AUS	1.3	1.1	0.7

- locale specific vocabulary halves the OoV rate

[a] Thanks Mark Paskin

Locale Matters (2)

Perplexity of unpruned LM:

Training Locale	Test Locale USA	GBR	AUS
USA	132	234	251
GBR	260	110	224
AUS	276	210	124

- locale specific LM halves the PPL of the unpruned LM

Locale Matters (3)

Perplexity of pruned LM:

Training Locale	Test Locale USA	GBR	AUS
USA	210	369	412
GBR	442	150	342
AUS	422	293	171

- locale specific LM halves the PPL of the pruned LM as well

Open Problems in Language Modeling for ASR and Beyond

- language model adaptation: bigger is not always better. Making use of related, yet not fully matched data, e.g.:
  - Web text should help query LM?
  - related locales—GBR, AUS should help USA?
- discriminative LM: ML estimate from correct text is of limited use in decoding, where the LM is presented with atypical n-grams (see lattice PPL experiment)
  - need parallel data (A, W^*) or not?
  - significant amount can be mined from voice search logs using confidence filtering

Google

ASR Success Story: Google Search by Voice

What contributed to success:

- excellent language model built from query stream
- clearly set user expectation by existing text app
- clean speech:
  - users are motivated to articulate clearly
  - app phones (Android, iPhone) do high quality speech capture
  - speech tranferred error free to ASR server over IP

Challenges:

- Measuring progress: manually transcribing data is at about same word error rate as system (15%)

Google

ASR Core Technology

Current state:

- automatic speech recognition is incredibly complex
- problem is fundamentally unsolved
- data availability and computing have changed significantly since the mid-nineties

Challenges and Directions:

- re-visit (simplify!) modeling choices made on corpora of modest size
- 2-3 orders of magnitude more data and computation is available
- multi-linguality built-in from start

Google