From sounds to words: Bayesian modelling of early language acquisition.

Sharon Goldwater

ilcc | Institute for Language, Cognition and Computation

THE UNIVERSITY of EDINBURGH informatics

Johns Hopkins Workshop: ‘Zero-Resource Speech Recognition’, July 2012

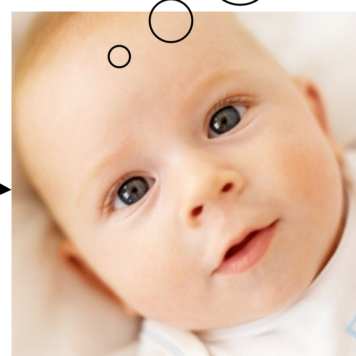
Language learning as induction

Input

(specific linguistic observations)

Grammar/lexicon

(abstract internal representation)



Output

(specific linguistic productions)

Constraints on learning

- Many generalizations are possible. What constrains the learner?
  - Innate constraints (domain-general or domain-specific).
  - Previously (or simultaneously) acquired knowledge: bootstrapping.

- How do these interact with each other and the input?
- How can we implement them in machines to improve coverage and accessibility of language technology?

Modeling approach

- Bayesian framework: a structured probabilistic approach.
  - Probabilistic: learner can exploit partial or uncertain information to help solve the bootstrapping problem.
  - Structured: models explicitly define representations, biases (constraints), and use of information.

Bayesian modeling

- An ideal observer approach.
  - What is the optimal solution to the induction problem, given particular assumptions about representation and available information?
  - In what ways might humans differ from this ideal learner, and why?

Outline

1. Introduction
2. Basic model: word segmentation from phonemic input
3. Lexical-phonetic learning from phonetic input
4. Word extraction from acoustic input

Word segmentation (idealized)

Input:

continuous
~~speech~~
text

lookatthedoggie
seethedoggie
shelookssofriendly
…

Output:

look at the doggie
see the doggie
she looks so friendly
…

segmented word tokens

at	doggie
the	friendly
she	look
so	looks
see	…

lexicon

Research questions

- Machine learning:
  - Can we develop a generative probabilistic model and effective inference method to discover the words?
  - I.e., can we learn an n-gram language model without knowing the words in advance?
- Cognitive science:
  - What kinds of assumptions must a learner make in order to discover words correctly?
  - Is a simple unigram model sufficient?

Bayesian learning

- Formulate the problem as a Bayesian model:

$$\underbrace{P(h|d)}_{\text{posterior}} \propto \underbrace{P(d|h)}_{\text{likelihood}} \underbrace{P(h)}_{\text{prior}}$$

- Focus is on the goal of the computation rather than on using a cognitively plausible learning algorithm.

Data:

lookatthedoggie
seethedoggie
shelookssofriendly
…

Hypotheses:

lookatthedoggie
seethedoggie
shelookssofriendly
…

l o o k a t t h e d o g g i e
s e e t h e d o g g i e
s h e l o o k s s o f r i e n d l y
…

look at thed oggi e
se e thed oggi e
sh e look ssofri e ndly
…

look at the doggie
see the doggie
she looks so friendly
…

$P(d|h)=1$

i like pizza
what about you
…

abc def gh
ijklmn opqrst uvwx
…

$P(d|h)=0$

Bayesian segmentation

- For segmentation,
  - Data: unsegmented corpus (transcriptions).
  - Hypotheses: sequences of word tokens.
- Under phonemic assumption, the prior does all the work.

$$\underbrace{P(h|d)}_{\text{posterior}} \propto \underbrace{P(d|h)}_{\text{likelihood}} \underbrace{P(h)}_{\text{prior}}$$

= 1 if concatenating words forms corpus,
= 0 otherwise.

Encodes assumptions of learner.

Unigram model

- Assume words are drawn from Dirichlet Process. Then

$$P(w_{n+1} = w \mid w_1 \ldots w_n) = \frac{n_w + \alpha P_0(w)}{n + \alpha}$$

with $P_0(w = x_1 \ldots x_m) = \prod_{i=1}^{m} P(x_i)$ for characters $x_1 \ldots x_m$.

  - "Rich-get-richer" process creates Zipfian distribution.
  - Base distribution P_0 favors shorter lexical items.
  - Number of lexical items grows with data size.
  - Probabilities don't depend on immediate context: unigram model.

Bigram model

- Assume words are drawn from hierarchical Dirichlet process.

$$P(w_{n+1} = w \mid w_n = w', w_1 \ldots w_{n-1}) = \frac{n_{(w',w)} + \beta P_1(w)}{n_{w'} + \beta}$$

$$P_1(w_{n+1} = w \mid w_1 \ldots w_n) = \frac{b_w + \alpha P_0(w)}{b + \alpha}$$

  - Similar assumptions to unigram model, but now words also depend on context.

Experiments

- Inference: simple Gibbs sampler.
  - Other methods possible (Mochihashi et al., 2009; Liang and Jordan 2010).
- Corpus:
  - 9790 utterances of phonemically transcribed child-directed speech (19-23 months).
  - Average 3.4 words/utterance, 2.9 phonemes/word.

```
yuwanttusiD6bUk
lUkD*z6b7wIThIzh&t
&nd6dOgi
yuwanttulUk&tDIs
...
```

Example results

Unigram model

```
youwant to see thebook
look theres aboy with his hat
and adoggie
you wantto lookatthis
lookatthis
havea drink
okay now
whatsthis
whatsthat
whatisit
look canyou take itout
...
```

Bigram model

```
you want to see the book
look theres a boy with his hat
and a doggie
you want to lookat this
lookat this
have a drink
okay now
whats this
whats that
whatis it
look canyou take it out
...
```

Conclusions

- Good segmentations of (phonemic) data can be found using fairly weak prior assumptions.
  - Utterances are composed of discrete units (words).
  - Units tend to be short.
  - Some units occur frequently, most do not.
  - Units tend to come in predictable patterns; i.e. context is key.

Further tests

- Frank, Goldwater, Griffiths, and Tenenbaum (2010):
  - Model captures human performance in an artificial segmentation task better than all other models tested.
- Pearl, Goldwater, and Steyvers (2010):
  - Incremental (but non-optimal) algorithms can sometimes yield more accurate results than batch versions.
  - Can “burstiness” be used to our advantage (as humans, or to design more efficient computer algorithms)?

Outline

1. Introduction
2. Basic model: word segmentation from phonemic input
3. Lexical-phonetic learning from phonetic input
4. Word extraction from acoustic input

Phones and words

- Most models of word segmentation use phonemic input.

(a)	intended:	/ju want wʌn/	/want e kʊki/
(d)	idealized:	/juwantwʌn/	/wantekʊki/

Phones and words

- Abstracts away from phonological and phonetic variation.

(a)	intended:	/ju want wʌn/	/want e kʊki/
(b)	surface:	[jə wãʔ wʌn]	[wan ə kʊki]
(c)	unsegmented:	[jəwãʔwʌn]	[wanəkʊki]
(d)	idealized:	/juwantwʌn/	/wantekʊki/

- But: phonological and word learning occur simultaneously and seem to interact.
  - How can we model this kind of joint learning? Will model predictions change?

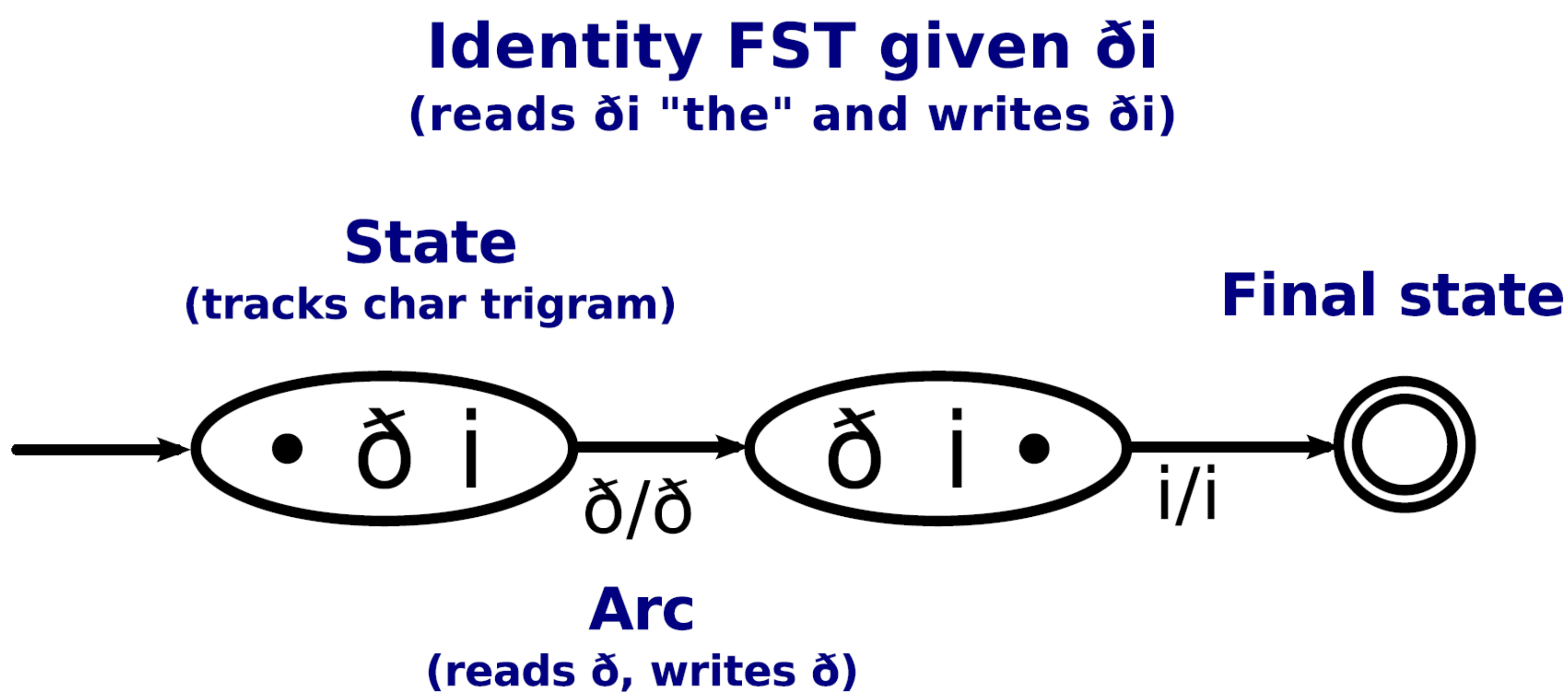
Joint learning

(a)	intended:	/ju want wʌn/	/want e kʊki/
(b)	surface:	[jə wãʔ wʌn]	[wan ə kʊki]

- Here: From surface forms, learn a lexicon, a language model, and a model of phonetic variation.
- Method: (unsupervised) noisy channel model.
  - Language model: similar to GGJ09.
  - Phonetic model: MaxEnt model using articulatory features.

Phonetic model

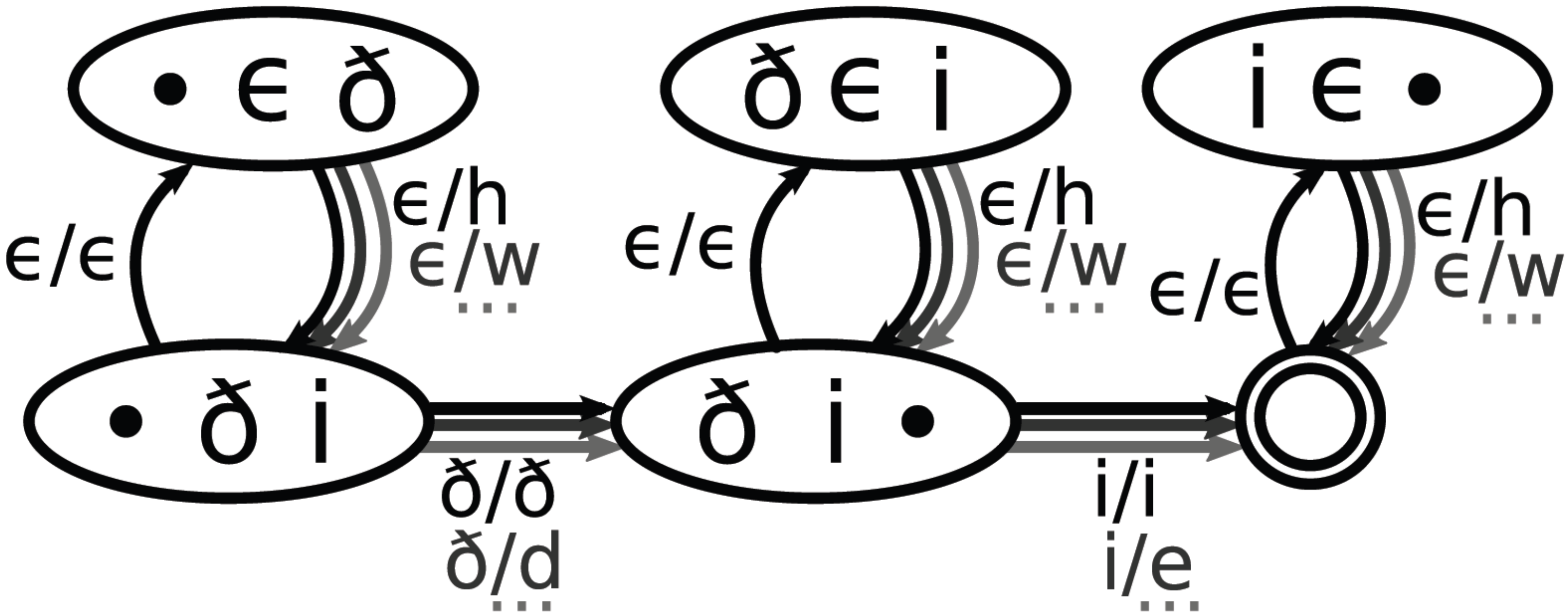
- Implemented as weighted finite-state transducer. Ex:



Our transducer

Reads ði, writes anything

(Likely outputs depend on parameters)



- Prob. of arc depends on features of sounds (same/different voicing/place/manner, etc.). Weights are learned.

Results so far (Elsner, Goldwater, and Eisenstein, 2012)

- Inference: approximate method greedily merges surface forms, retrains transducer after each merging pass.
- Data: simulate phonetic variation in BR corpus by sampling phonetic forms from Buckeye corpus.

"about"	ahbawt:15, bawt:9, ihbawt:4, ahbawd:4, ih-bawd:4, ahbaat:2, baw:1, ahbaht:1, erbawd:1, bawd:1, ahbaad:1, ahpaat:1, bah:1, baht:1, ah:1, ahbahd:1, ehbaat:1, ahbaed:1, ihbaht:1, baot:1

Results so far (Elsner, Goldwater, and Eisenstein, 2012)

- Inference: approximate method greedily merges surface forms, retrains transducer after each merging pass.
- Data: simulate phonetic variation in BR corpus by sampling phonetic forms from Buckeye corpus.
- Results:

	Token F	Lexicon F
Baseline	.65	.67
Unigram LM	.75	.76
Bigram LM	.79	.87

What about segmentation?

- System also improves lexicon when using inferred word boundaries (from GGJ09).
- But:
  - Overall performance much worse (.44 → .49 vs. .65 → .79).
  - Iterating segmentation and lexicon learning doesn't help.
- In progress: new system with beam sampler instead of greedy search, simultaneously learns segmentation.

Conclusions

- First joint model of phonetic and word learning using word-level context info on phonetic corpus data.
- Additional evidence that word and phone learning can inform each other.
- As in phonemic model, word-level context is important—helps disambiguate similar-sounding words (e.g., what/wet).
- Dealing with segmentation ambiguity also is hard.

Outline

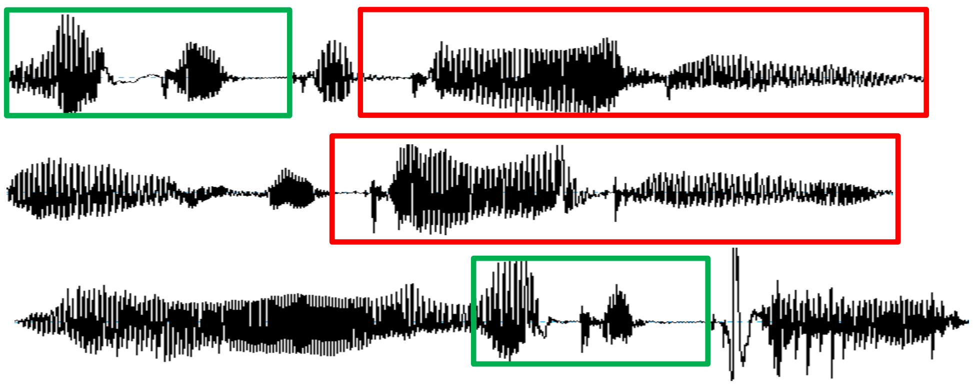
1. Introduction
2. Basic model: word segmentation from phonemic input
3. Lexical-phonetic learning from phonetic input
4. Word extraction from acoustic input

Learning words from acoustics

- Goal: investigate incremental (online) learning in a model of whole-word extraction from speech.
- Method: modify Park and Glass (2008) algorithm to be (more) incremental.

Algorithm

1. Compare pairs of utterances to extract pairs of acoustically similar speech fragments:



Look at the doggie

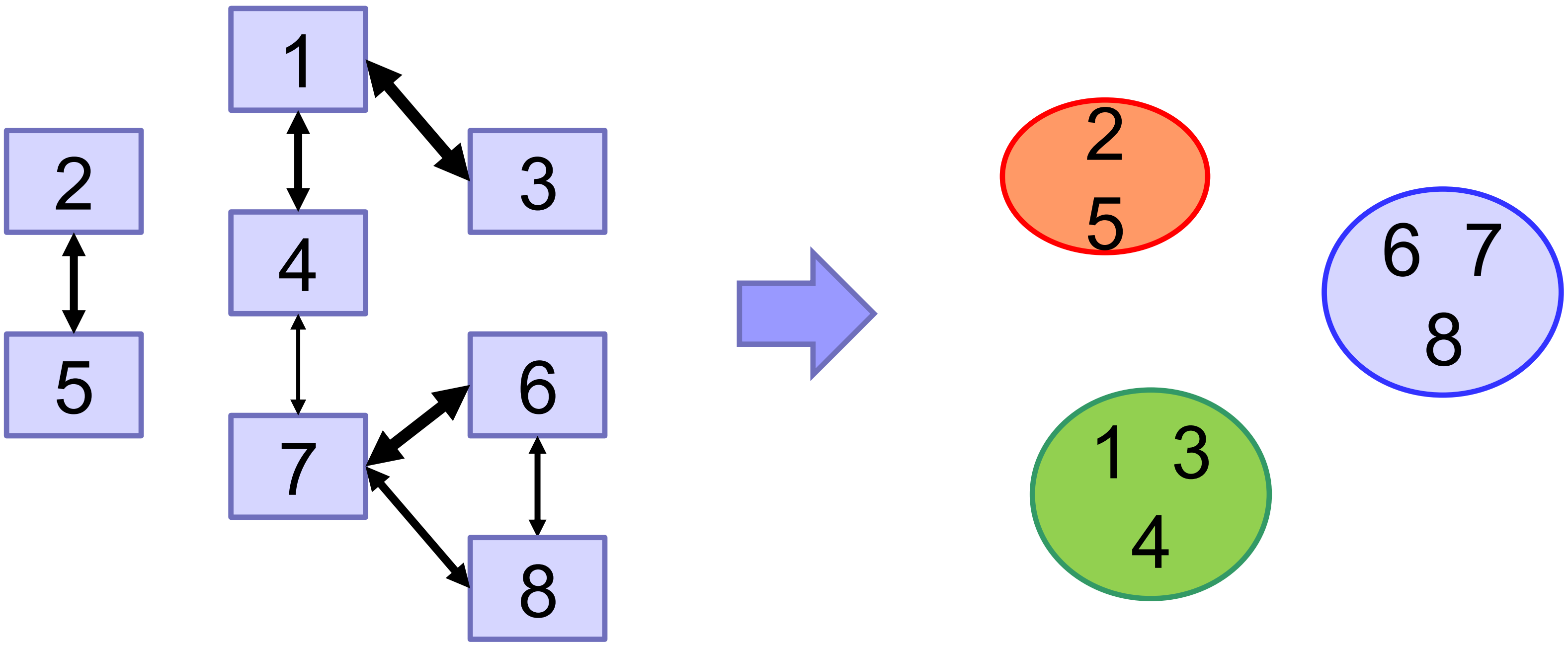
Where's the doggie

Yeah, look at that

- Uses a slightly modified version of P&G's Segmental DTW.
- Compares only with fixed-size window of utterances.

Algorithm:

2. Cluster together extracted fragments pairs into larger groups: lexical items.



Experiments:

- Test on recordings from parents of 9-15 month-olds.
- Measure *entropy reduction* and examine words found.
- Results:
  - Original corpus: Little difference between using limited window (10-20 utterances) or full batch mode.
  - Permuted corpus: Limited window results are much worse.
  - ‘Mommy’ and child’s name are found in most sessions.

Conclusions

- Frequent nearby repetitions are helpful to the incremental learner: limited memory is almost as good as batch learning.
- Simple pattern-matching can extract word-like units, but boundaries not always accurate.
- Open issues:
  - Online clustering method.
  - Relationship between these units and sub-word units.
  - Word extraction vs. word segmentation.

Issues/ideas for future work

- Extending Bayesian models to work from acoustics.
  - Lee & Glass (2012) a great start on phonetic clustering; can we build in higher-level dependencies and learn a lexicon?
  - Consider intermediate levels of representation (syllables).
- Developing better inference methods.
  - Efficient for machine learning; cognitively plausible for human models.
  - Can we exploit structure that isn't captured by our generative models (e.g., burstiness) to design better inference methods? (Or should we design more accurate models?)
- Using non-acoustic information (e.g., articulatory gestures, visual context, gesture).

Bayesian model

Assumes word w_i is generated as follows:

1. Is w_i a novel lexical item?

$$P(yes) = \frac{\alpha}{n + \alpha}$$

$$P(no) = \frac{n}{n + \alpha}$$

Fewer word types = Higher probability

Bayesian model

Assume word w_i is generated as follows:

2. If novel, generate phonemic form $x_1 \ldots x_m$:

$$P(w_i = x_1 \ldots x_m) = \prod_{i=1}^{m} P(x_i)$$

Shorter words = Higher probability

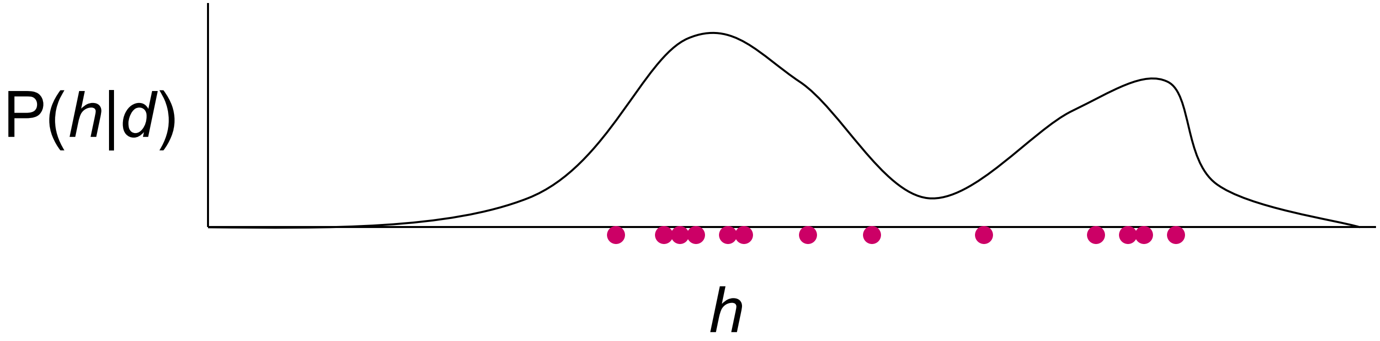
If not, choose lexical identity of w_i from previously occurring words:

$$P(w_i = w) = \frac{n_w}{n}$$

Power law = Higher probability

Learning algorithm

- Model defines a distribution over hypotheses. We use Gibbs sampling to find a good hypothesis.
  - Iterative procedure produces samples from the posterior distribution of hypotheses.



  - A batch algorithm, assumes perfect memory for data.

Inference

- We use a Gibbs sampler that compares pairs of hypotheses differing by a single word boundary:

```
whats.that
the.doggie
yeah
wheres.the.doggie
…
```

```
whats.that
the.dog.gie
yeah
wheres.the.doggie
…
```

- Calculate the probabilities of the words that differ, given current analysis of all other words.
- Sample a hypothesis according to the ratio of probabilities.

Incremental Sampling

For each utterance:
- Sample a segmentation from the posterior distribution given the current lexicon.
- Add counts of segmented words to lexicon.

- Online algorithm
- Limits memory for corpus data

(Particle filter: more particles ⇔ more memory)

Testing model predictions

- Saffran-style experiment using multiple utterances.
  - Synthesize stimuli with 500ms pauses between utterances.

lagi dazu tigupi bavulu kabitudu kipavazi

→

lagitigupibavulukabitudulagikipavazi dazukipavazibavululagitigupikabitudu kipavazitigupidazukabitudulagitigupi …

  - Training: adult subjects listen to corpus of utterances.
  - Testing: 2AFC between words and part-word distractors
- Compare our model (and others) to humans, focusing on changes in performance as task difficulty is varied.

Experiment 1: utterance length

- Vary the number of words per utterance.

#vocab	# wds/utt	# utts	tot # wds
6	1	1200	1200
6	2	600	1200
6	4	300	1200
6	6	200	1200
6	8	150	1200
6	12	100	1200

Experiment 2: exposure time

- Vary the number of utterances heard in training.

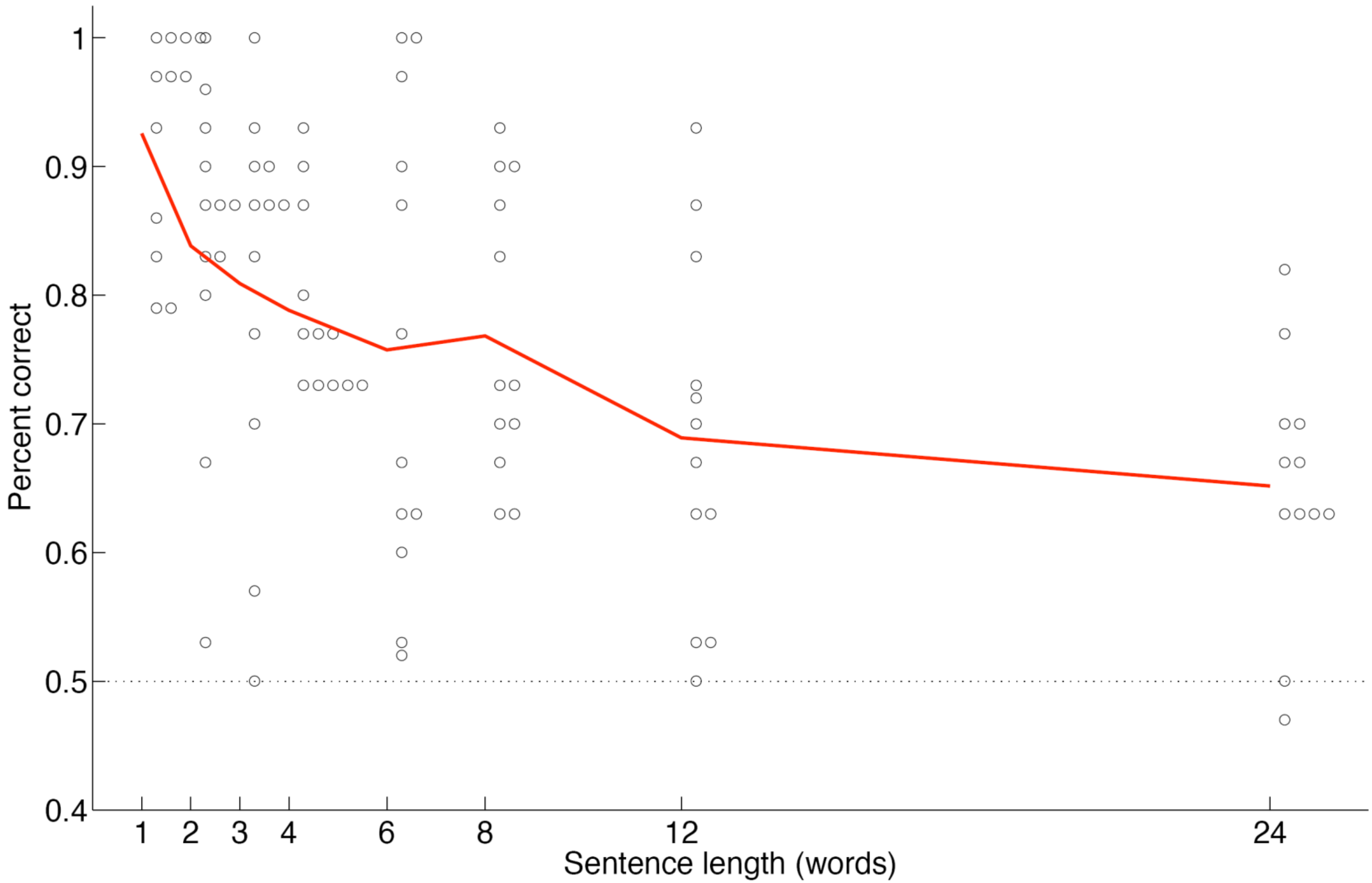
#vocab	# wds/utt	# utts	tot # wds
6	4	12	48
6	4	25	100
6	4	75	300
6	4	150	600
6	4	225	900
6	4	300	1200

Experiment 3: vocabulary size

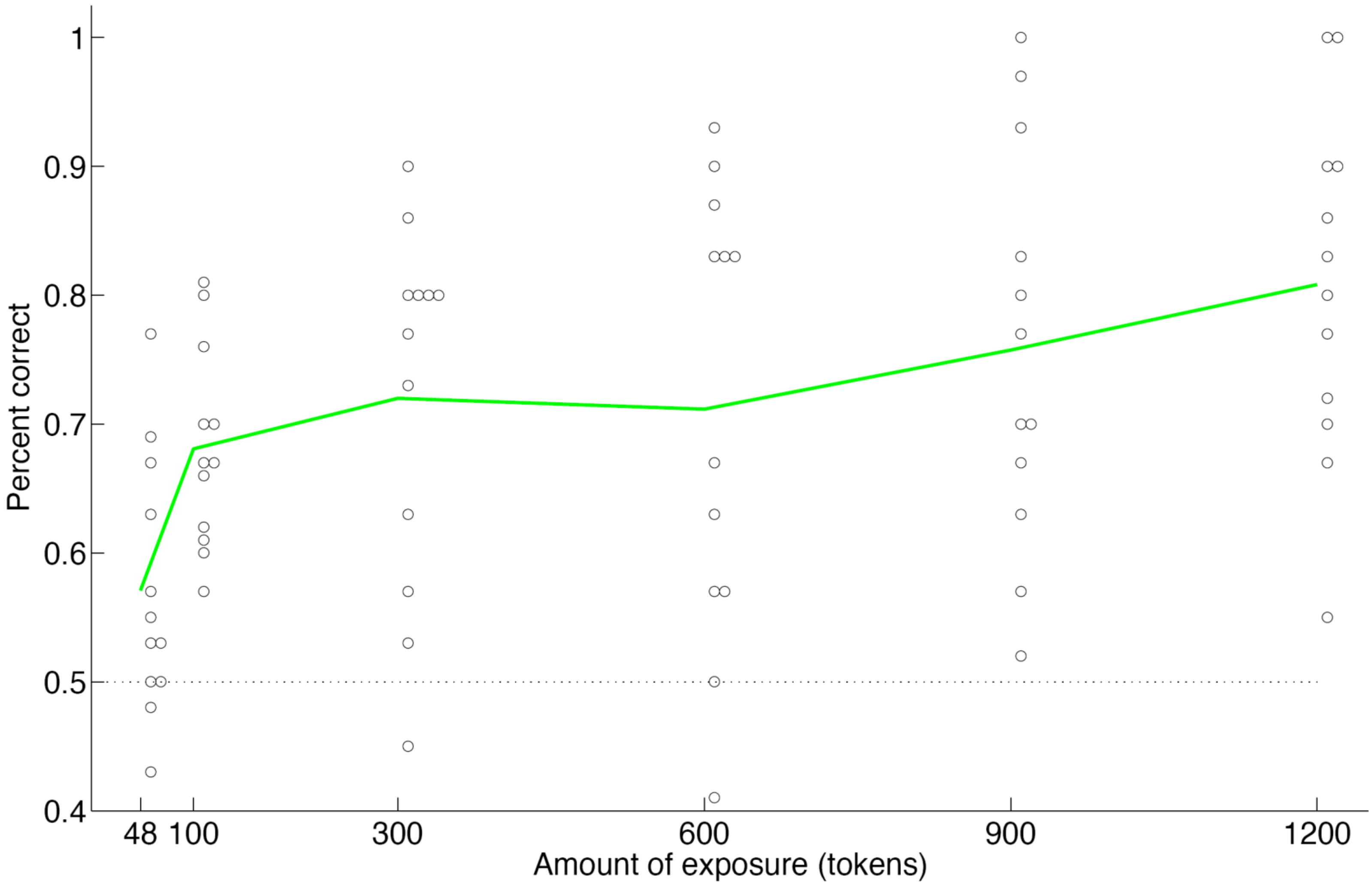
- Vary the number of lexical items.

#vocab	# wds/utt	# utts	tot # wds
3	4	150	600
4	4	150	600
5	4	150	600
6	4	150	600
9	4	150	600

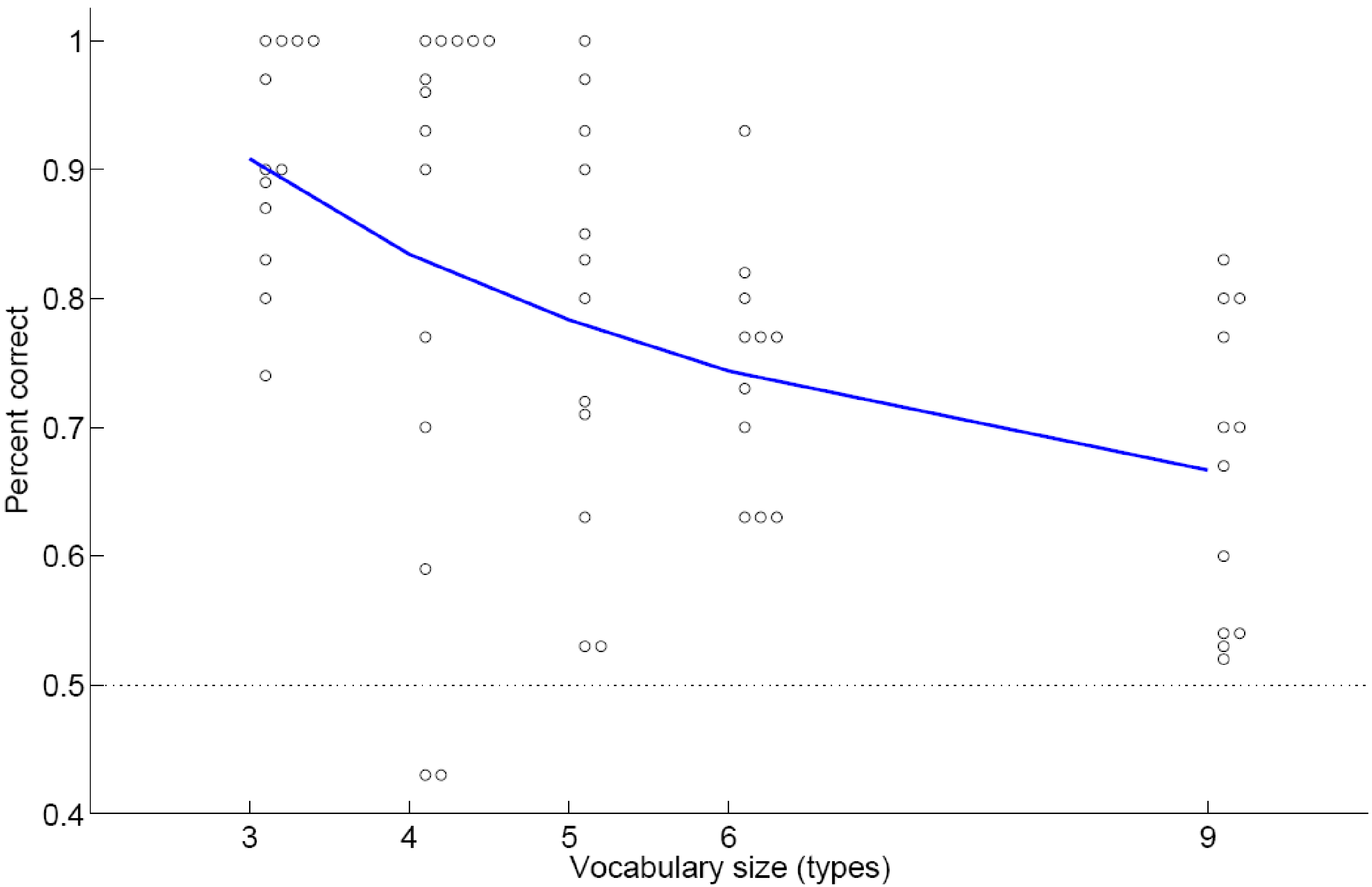
Human results: utterance length



Human results: exposure time



Human results: vocabulary size



Model comparison

- Evaluated six different models.
- Each model trained and tested on same stimuli as humans.
- For testing, produce a score $s(w)$ for each item in choice pair and use Luce choice rule:

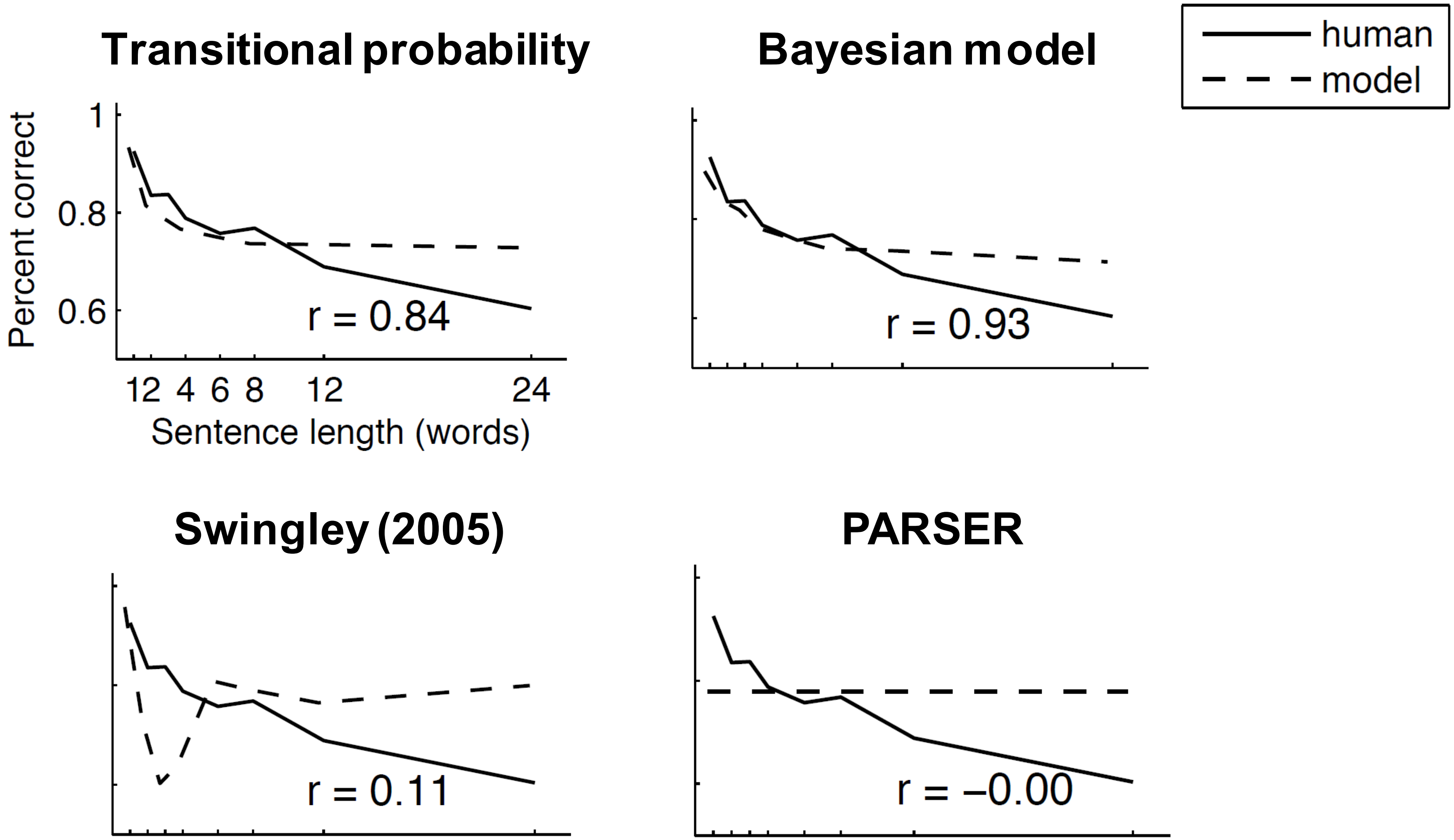
$$P(w_1) = \frac{s(w_1)}{s(w_1) + s(w_2)}$$

- Calculate correlation coefficients between each model's results and the human data.

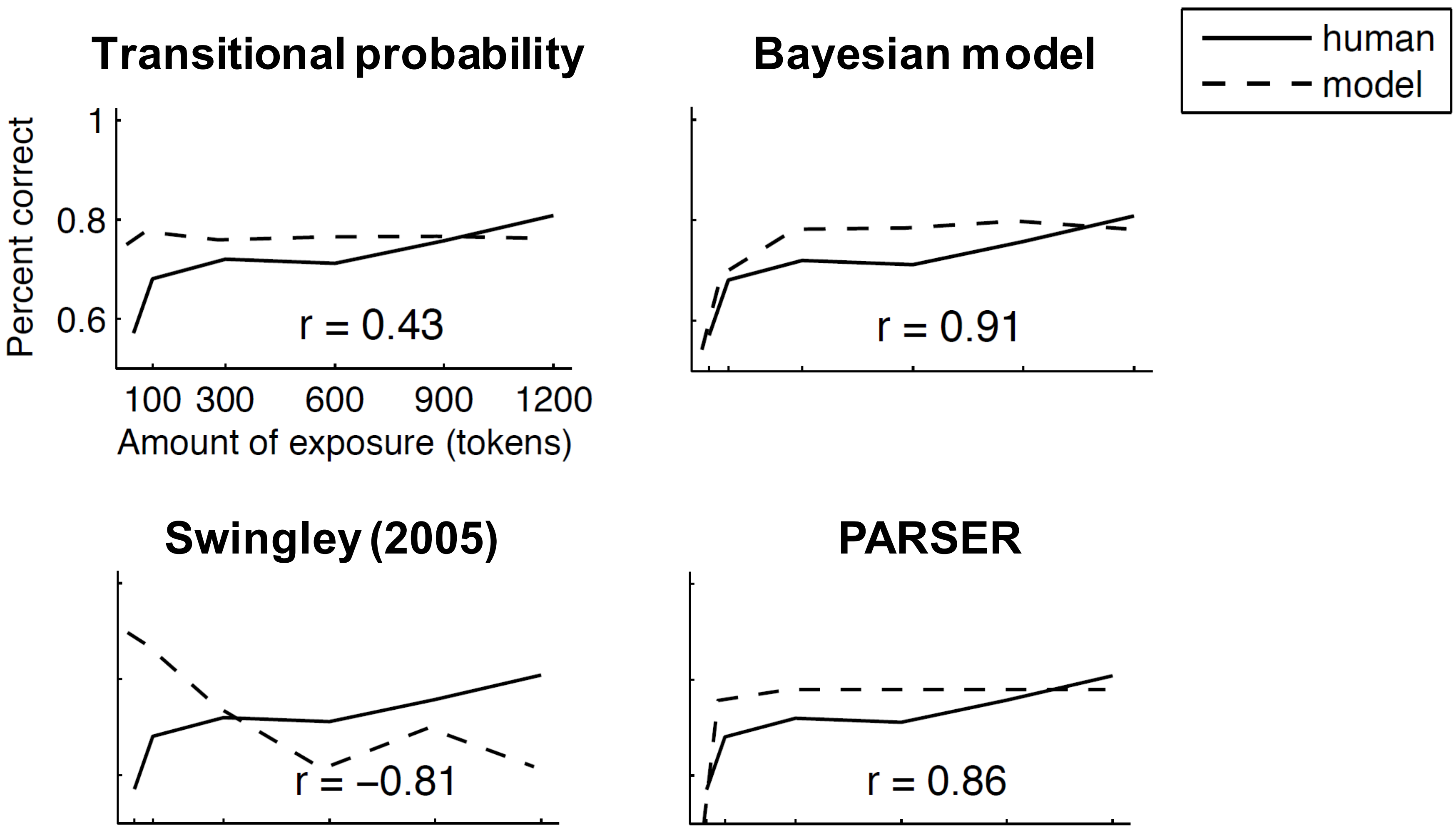
Models used

- Several variations on transitional probabilities (TP)
  - $s(w)$ = minimum TP in w.
- Swingley (2005)
  - Builds lexicon using local statistics and frequency thresholds.
  - $s(w)$ = max threshold at which w appears in lexicon.
- PARSER (Perruchet and Vintner, 1998)
  - Incorporates principles of lexical competition and memory decay.
  - $s(w) = P(w)$ as defined by model.
- Bayesian model
  - $s(w) = P(w)$ as defined by model.

Results: utterance length



Results: exposure time



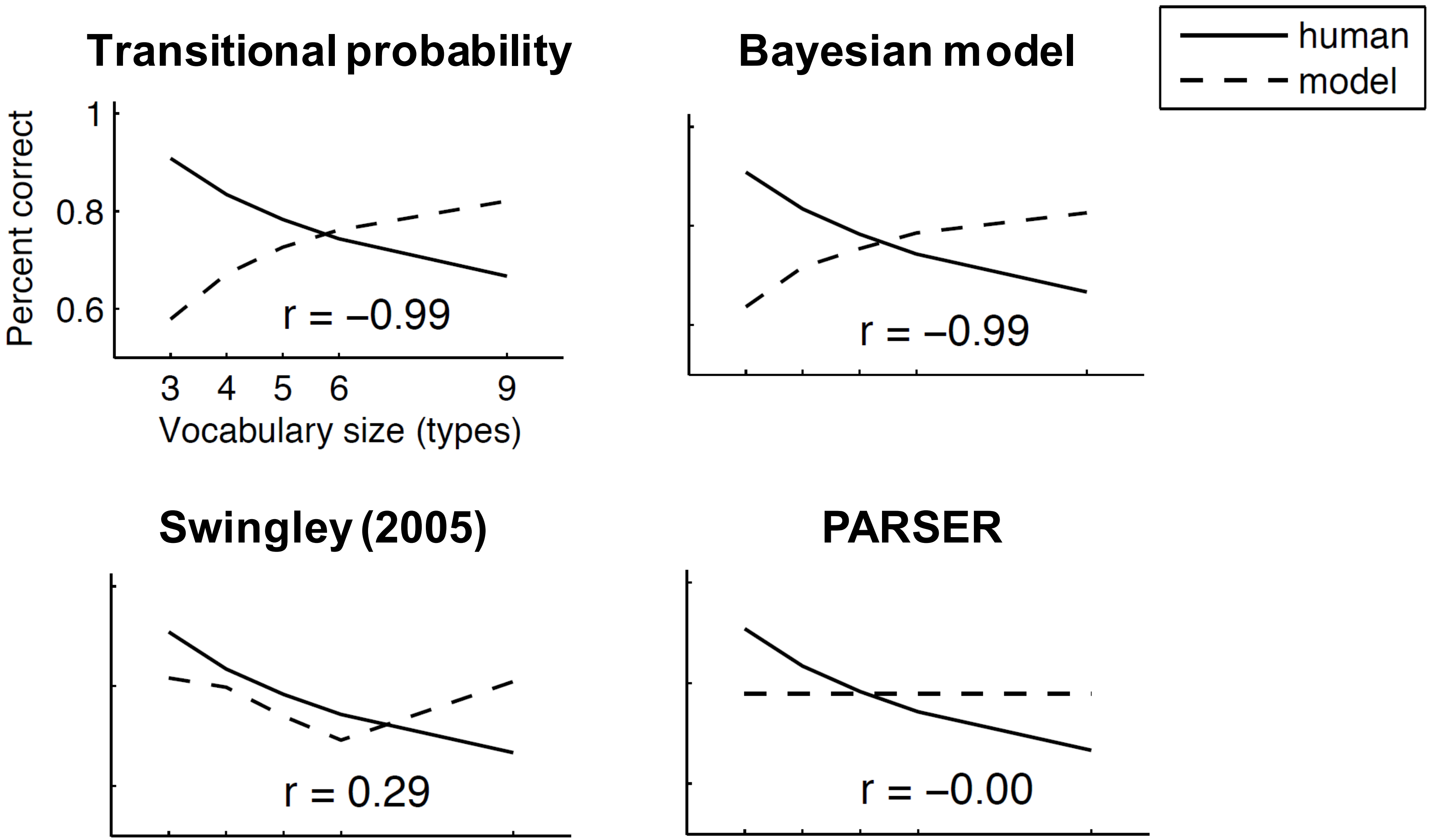
Summary: Experiments 1 and 2

- For humans, learning to segment is more difficult
  - when utterances contain more words.
  - when less data is available.
- Only Bayesian model captures both effects:

	TPs	Sw05	PARSER	Bayes
Utt length	✓	×	×	✓
Exposure	×	×	✓	✓

- Success is due to accumulation of evidence for best hypothesis, moderated by competition with other hypotheses.

Model results: vocabulary size



What's going wrong?

- TPs: smaller vocab => TPs across words are higher.
- Bayes: smaller vocab => Incorrect solutions have relatively small vocabularies with many frequent "words".

lagitigupi kabitudulagi
tigupi lagi kabitudulagi
kabitudulagi kabitudu tigupi
lagi kabitudu lagitigupi
kabitudulagi tigupi kabitudu
…

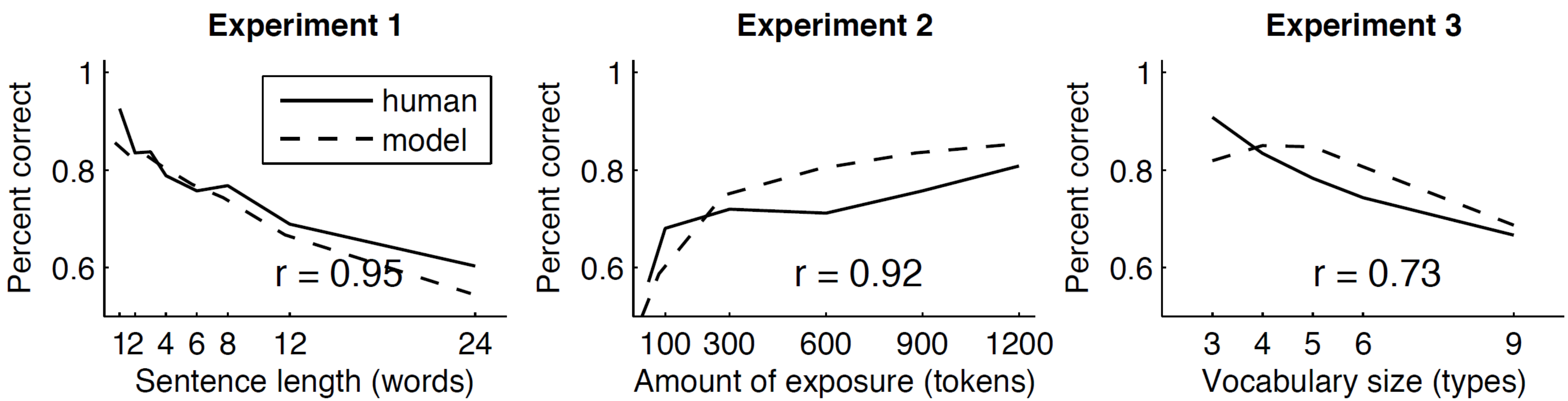
- With perfect memory, stronger statistical cues of larger vocabulary outweigh increased storage needs.

Memory limitations

- Modified Bayesian model has limited memory for data and generalizations.
  - Online learning algorithm processes one utterance at a time, one pass through data.
  - Random decay of items in lexicon.
- Learner is no longer guaranteed to find optimal solution.

Results: memory-limited learner

- Good fit to all three experiments:



- Simulating limited memory in TP also improves results but not as much.

Summary

- Humans behave like ideal learners in some cases.
  - Longer utterances are harder – competition.
  - Shorter exposure is harder – less evidence.
- Humans are unlike ideal learners in other cases.
  - Larger vocabulary is harder for humans, easier for model.
- Memory-limited learner captures human behavior in all three experiments.

Targets vs. distractors