Symbolic Constraints and Statistical Methods: Use Together for Best Results

Constantine Lignos

University of Pennsylvania
Department of Computer and Information Science
Institute for Research in Cognitive Science

Johns Hopkins Human Language Technology
Center of Excellence

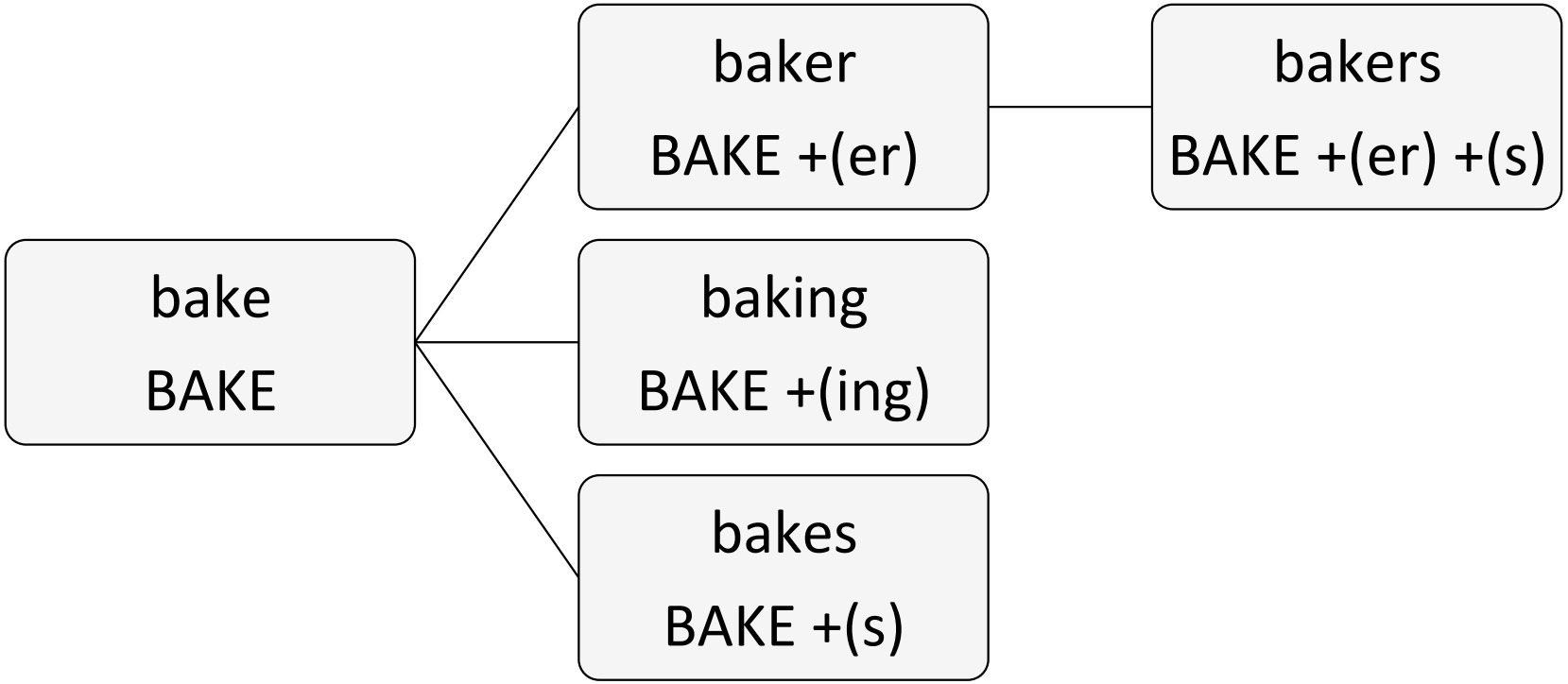
2/18/2013

I. Introduction

My focus: Computationally efficient solutions to hard natural language problems.

Insane Heuristic (Siklóssy, 1974):
"To solve a problem, solve a harder problem."

Morphology learning: MORSEL



Unsupervised learning of morphological structure, state-of-the-art for Finnish and English.
https://github.com/ConstantineLignos/MORSEL

(BUCLD 2009; Morpho Challenge 2009, best paper 2010; Research on Language and Computation, 2010)

Symbolic constraints and statistical methods

- With the right constraints on the learning problem, simple learning strategies work
- Where should these constraints come from?
- Common practice: error-motivated
  - Posterior regularization (Graça et al., 2011): Constrain POS tags to have a verb in every sentence, sparse tag distribution
  - Constrain word segmenter to put a vowel in every word (Venkataraman, 2001)
- I suggest: all language comes from the mind; search within its constraints.

II. Infant word segmentation

Word segmentation is an *unsexy* problem.

Whereareyougoing?
Howdoesabunnyrabbitwalk?
Doeshewalklikeyouordoeshegohophophop?
Whatareyoudoing?
Sweepbroom.
Isthatabroom?
Ithoughtitwasabrush.

Adam's mother (Brown, 1973)

Where are you going?
How does a bunny rabbit walk?
Does he walk like you or does he go hop hop hop?
What are you doing?
Sweep broom.
Is that a broom?
I thought it was a brush.

Adam's mother (Brown, 1973)

Modeling of WS

- Transitional probability minima (Saffran, 1996 et seq.; Swingley, 2005) by themselves are a dead-end (Yang, 2004)
- Likewise for phonotactics-only accounts (Daland and Pierrehumbert, 2011)
- Models using lexicon, and Bayesian inference (Borschinger and Johnson, 2011; Goldwater et al., 2009; Johnson and Goldwater, 2009; Pearl et al., 2011)
  - Goal of learner is develop high-quality lexicon for segmentation, context can be crucial
  - Integrates progress in unsupervised learning (Teh, 2006)
  - High performance. Developmental impact?

Marr's three levels of description/analysis

- Computational: system's goal
  - What problem does the system solve? Input/output constraints
- Algorithmic: system's method
  - How does the system do what it does: details of processes and representations used
- Implementational: system's means
  - How is the system physically realized?

Connecting to development

- Can we connect these models to development?
  - Primarily models at *computational* level, not *algorithmic* (Marr, 1983)
- What about the how?
- My interest: how can we explain the behavior of children as they learn to segment words?

Our modeling goal:

Build the simplest model that:

- Aligns with infants' capabilities
- Replicates infants' behavior in a principled fashion
- Performs reasonably at the task

How do infants segment speech?

- Possible strategy: identification of words in isolation (Peters, 1983; Pinker et al., 1984)
  - Unlikely to be sufficient (Aslin et al., 1996), but probably helpful (Brent and Siskind, 2001)
- Attending to multiple cues in the input, most popularly:
  - Bootstrapping from known words (Bortfeld et al., 2005; Dahan and Brent, 1999)
  - More easily identify novel words at beginning and ends of utterances at 8 months (Seidl & Johnson, 2006)
  - Dominant stress pattern of language (Jusczyk et al., 1999)

Infant word learning

- Infants show first clear signs of identifying words at 6 mos. (Bergelson and Swingley, 2012)
- At that age, no:
  - Stress preference (Jusczyk et al., 1993, 1999)
  - Phoneme transition preference/phonotactics (Mattys et al., 1999)
- But:
  - Syllable as primary perceptual unit, sensitivity over time:
    - Birth: number of syllables (Bijeljac-Babic et al., 1993); 2 mos: preference for syllabifiable input (Bertoncini & Mehler, 1981), holistic syllables (Jusczyk & Derrah, 1987)
  - Ability to identify cohesive chunks (Goodsitt et al, 1993)

Overview of the proposed algorithm

- Segmenter has a lexicon of potential words it builds over time
  - Starts empty, words are added based on segmentation of each utterance
  - Each word has a score
- Use words in the lexicon to divide an utterance by *subtractively segmenting* from the front of the utterance
- Operates online
  - Processes one utterance at a time
  - Cannot remember previous utterances or how it segmented them, only lexicon
- Operates left-to-right in each utterance to insert word boundaries between syllables

Useful constraints

- Like infants, start work from the outside in
- Restrict the search for possible segmentations to words we already know when possible
- Use simple reinforcement learning to reward useful words and penalize bad ones

- I'll work through some examples
  - Orthography for easy reading, input is syllabified phonemes

In the beginning…

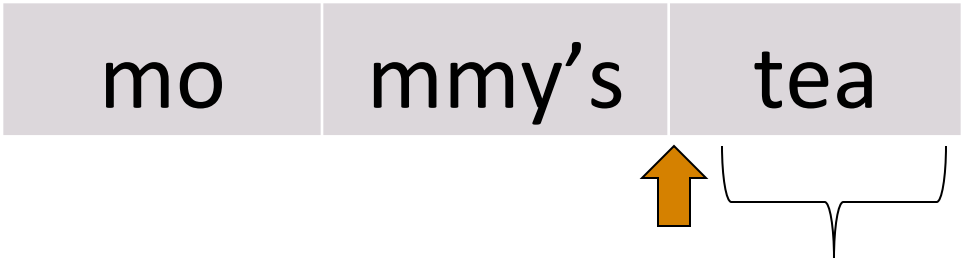
- Just add whole utterances to the lexicon
- Gets words in isolation for free, but often more than one word

Lexicon:
bigdrum

Subtractive Segmentation

- Use words in the lexicon to break up the utterance
- Increase word's score when it is used
- Add new words to lexicon

Lexicon:
mommy's
tea
…

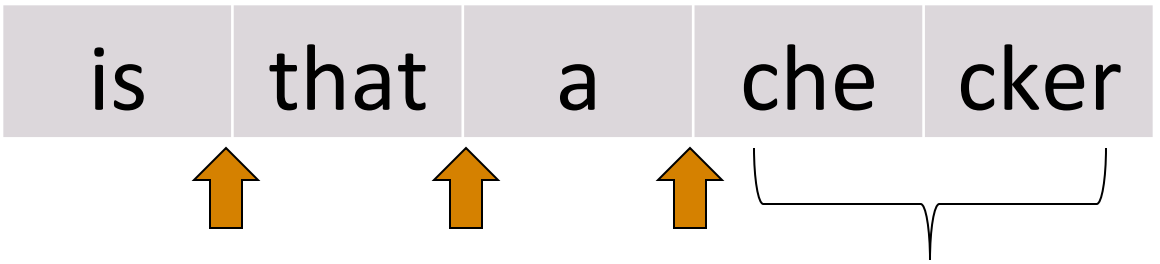


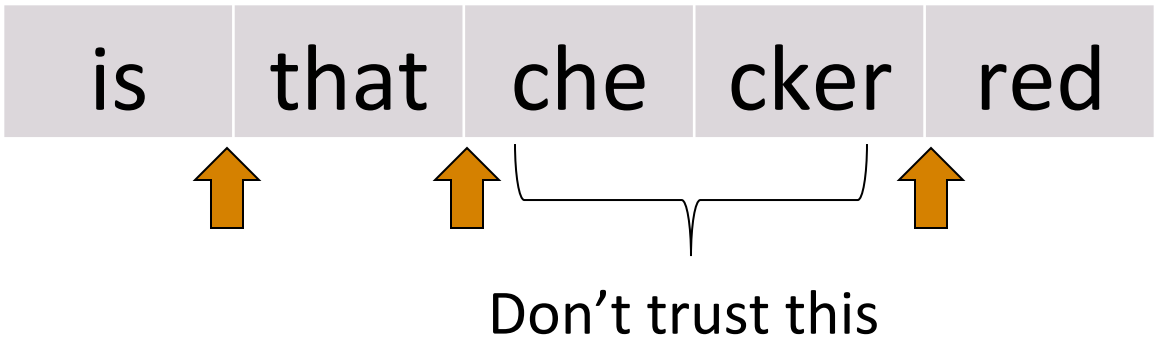
Treat remainder as word, add to lexicon

Trust

- Add new words to lexicon based on whether we *trust* them (touch an utterance boundary)

Lexicon:
a
is
that
red
checker
…

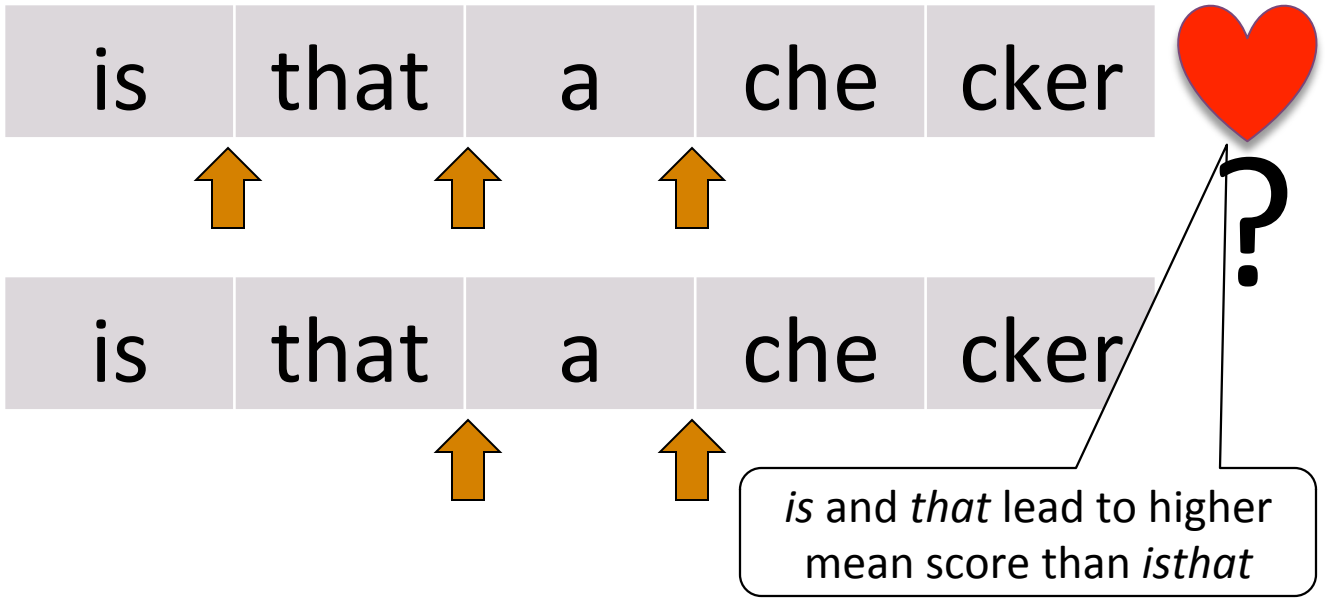




Multiple hypotheses

- For multiple possible subtractions two options:
  - Greedy approach (Lignos and Yang, 2010)
  - Pursue two (or more) hypotheses (*beam search*)
- Two hypotheses allow for *penalization*: reduce score of word that started losing hypothesis

Lexicon:
a
is
isthat
that
…



Scoring hypotheses

- Prefer the hypothesis that uses the higher-scoring words
  - Winner is rewarded, word scores will go up: “rich get richer”
- Geometric mean of scores of words used as metric:

$$\arg\max_{H}\left(\prod_{w_i \in H} score(w_i)\right)^{\frac{1}{n}}$$

  - Useful for compound splitting (Koehn and Knight, 2003; Lignos, 2010)
  - Doesn’t have bias for fewer or more words, but seeks a higher average score
  - Post-hoc testing shows this outperforms other obvious metrics (MDL/MLE, other means) (ask me)

Predictions

- Default assumption of utterance = word →
  Infants will start with oversized units and words in isolation
- Rich-get-richer scoring →
  As the learner is exposed to more data, learner will tend to use high-frequency elements
- Penalization →
  Use of collocations will decrease with time

Our evaluation corpus

- Constructed from Brown (1973) subset of CHILDES English (Adam, Eve, Sarah), ~60k utterances
- Pronunciations and stress for each word from CMUDICT, algorithmically syllabified
- Sample input:

B.IH.G|D.R.AH.M
HH.AO.R.S
HH.UW|IH.Z|DH.AE.T

Evaluation

- A' calculated over syllable boundaries
  - Balance of hit rate and false alarm rate for discriminating word boundaries
  - Trapezoidal approximation of area under ROC curve
- F-score used for word token identification
  - Balance of precision (how often a word identified in an utterance is correct) and recall (how many correct words were found)
  - Ex: *is that a lady* segmented as *isthat a lady*
    - Precision 2/3: *a, lady*; *isthat*
    - Recall 2/4: *a, lady; is, that*

Evaluation

- Evaluated segmenter in three forms:
  - Subtractive segmentation
  - Subtractive segmentation with *trust*, only adding words to the lexicon if they touch an utterance boundary
  - Subtractive segmentation with trust and *multiple hypotheses*, considering two hypothetical segmentations and penalizing the loser
- Errors computed over testing corpus from first utterance
  - Worst-case evaluation for online learner: **zero training**.
- Word boundaries taken as orthographic boundaries in the input
  - Aligns with morphological and phonological definitions of word

Performance

Method	Hit	False Alarm	A’	Word F-Score
Baseline: syllable = word	1.0	1.0	0	0.753
Subtractive Segmentation	**0.992**	0.776	0.795	0.797
+Trust	0.961	0.468	0.860	0.841
+Multiple Hypotheses	0.953	**0.401**	**0.875**	**0.849**

- Trust and multiple hypotheses significantly reduce FA rate
- Perfect memory is not crucial: evaluating A’ and F-score with imperfect memory yield similar results
- Segments 60k utterances in < 1 second
- Beam performs *better* than exhaustive search (ask me)

Infant error patterns

- Undersegmentation at young age (Brown, 1973; Clark, 1977; Peters, 1977, 1983)
  - Function word collocations: *that-a*, *it's, isn't-it*
  - Phrases as single unit:

    look at that, what's that, oh boy, all gone, open the door
  - Function-content collocations: *picture-of*, *whole-thing*
- Oversegmentation at older ages (Peters, 1983)
  - Function word oversegmentation: *behave/be have*, *tulips/two lips, Miami/my Ami/your Ami*
  - Errors can still occur in adulthood (Cutler and Norris, 1988)

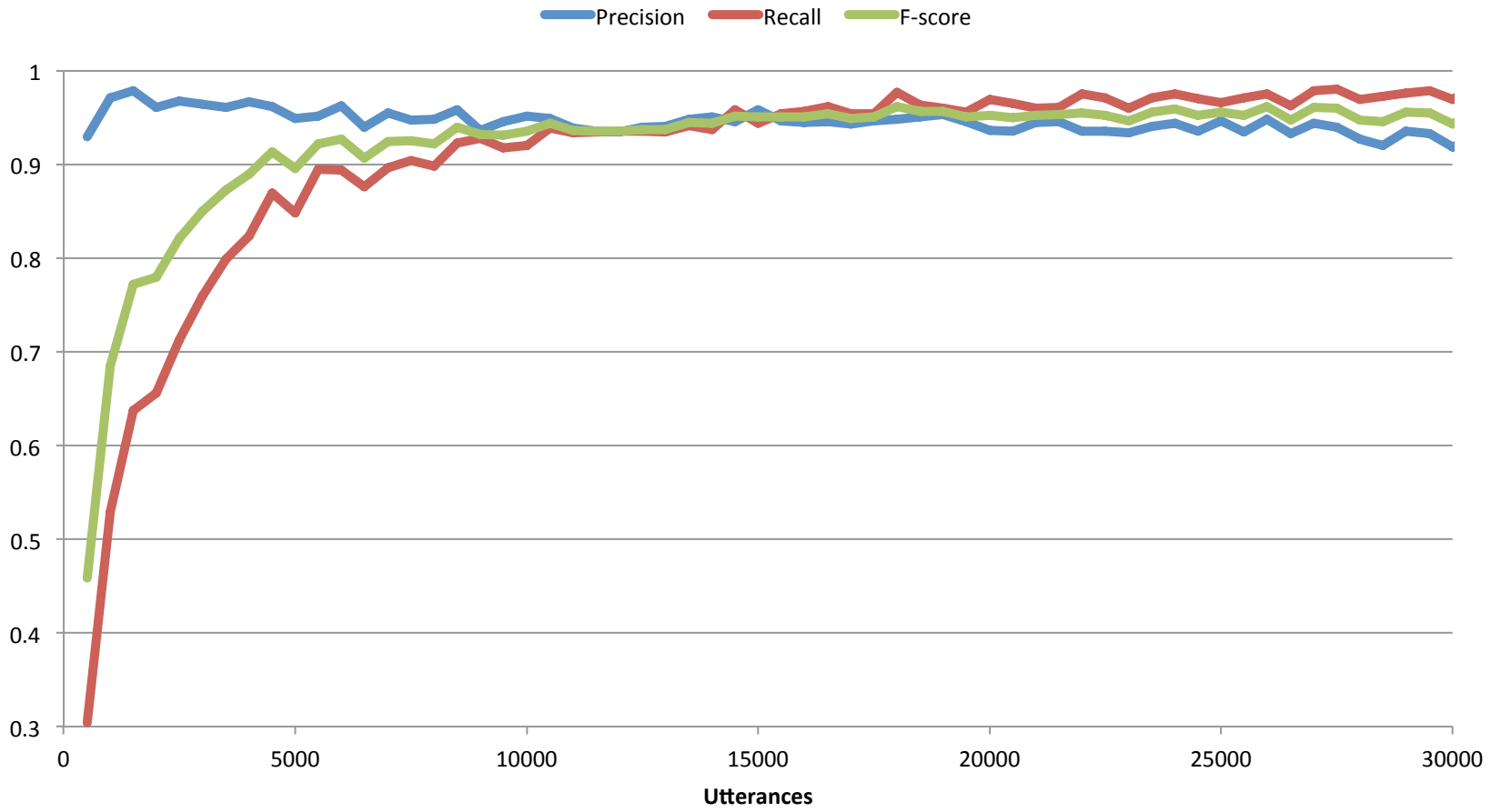
Most frequent error tokens

- Divided into early (first 10k utterances) and late (last 10k utterances) stages of learning
- Coded most frequent incorrect words in output as:
  - Function: Overuse of function word (away→ a way)
  - Function collocation: Two function words (that's a → that'sa)
  - Content collocation: Content and content/function word (a ball → aball)
  - Other
- Distribution changes across time (Chi-squared $p < .0001$)

Time	Function	Func. Colloc.	Cont. Colloc.	Other
Early	44.2%	37.0%	8.5%	10.4%
Late	70.6%	1.0%	1.2%	27.3%

Learning curve

- Learner starts undersegmenting, as it learns achieves balance with slight oversegmentation



Extensions: Joint morphology learning

- Learned on output of segmentation
- Initial lexicon of high-enough quality to learn common suffixes
- Acquisition order similar to Brown 1973:

Predicted order	Suffix	Example	Brown Average Rank
1	-ɪŋ	bake-ing	2.33
2	-z	happen-s	7.9
3	-d	happen-ed	9
4	-ɚ	bake-er	-
5	-t	check-ed	9
6	-ən	broke-en	-
7	-i:	smell-y	-

Word segmentation: Conclusions

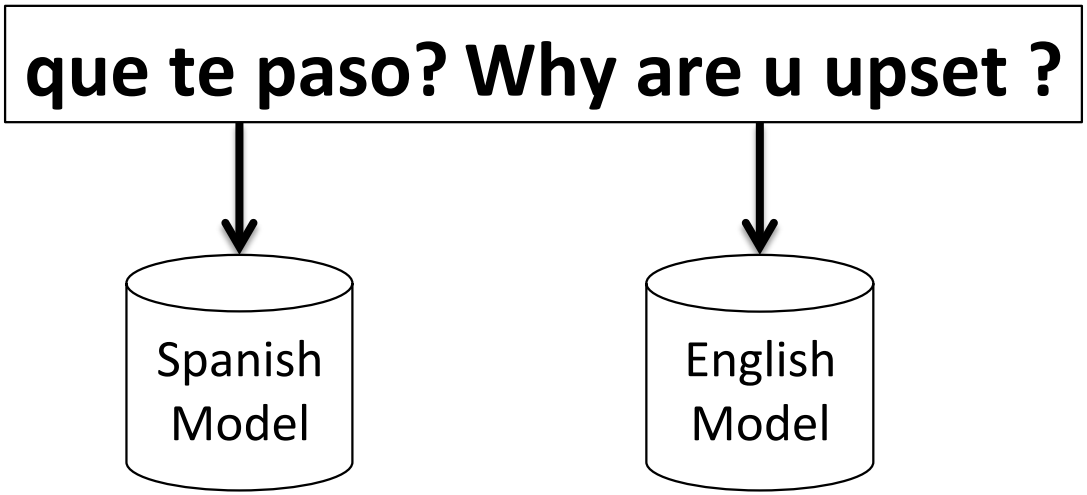
- By working at the algorithmic level, we can use informational from experimental results to build systems
- Taking advantage of what we know about infant behavior (syllables, segmentation strategies) leads to good performance and an efficient algorithm
- A simple reward-based approach predicts the learning patterns of infants

III. Codeswitching

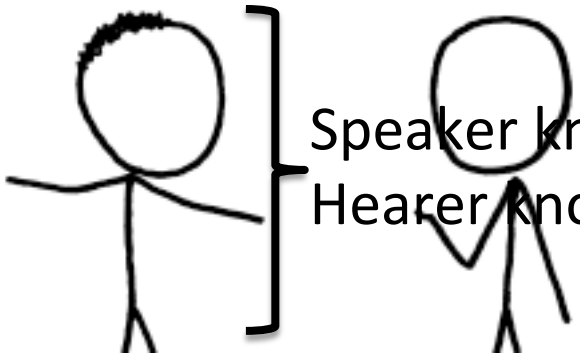
i'm at work babe. trabajando like a good girl. que haces. estoy entusiamada porque voy al mall con gina!!!

I have a bit of a situation mañana marcame porfas, creo que no voy a tener coche!

Value of detecting codeswitching



- Create single-language chunks for normalization/MT:
  - "**y** u no…" vs. "manzanas **y** bananas"
- Characterize communicants using language knowledge



Challenges

- Only existing codeswitching corpora are interview-scale
- Assume no text normalization or part-of-speech tagging
- Short messages, short phrases
- Technical terms (“USB”) and brand names (“Apple”)

Assumptions

- We can collect corpora *primarily* consisting of each language of interest
  - Collections of primarily Spanish (6.8m) and English (2.8m) tweets
- Try to make fewest assumptions about match between training data and applications
  - May not be same medium (e.g., Twitter, SMS)
  - May not have same dominant language in the pair
  - *No annotated codeswitched data for training*

Codeswitchador methodology

- Label each token using source language
- Threshold on number of hits to perform language identification and codeswitching
  - Require 2 tokens for a language to call it present
  - If multiple languages present, label it codeswitched

Models 1 and 1.5

- Language ratio:

$$\frac{\log(P(w \mid Spanish))}{\log(P(w \mid English))}$$

- Model 1: Label words of ratio above 1 Spanish, below as English (MLE)

- *Intuition* for ratio ≈ 1 *:*
  - *S*yntactic context constraints codeswitching (Belazi et al., 1994)
  - Syntactic heads for English/Spanish on left

- Model 1.5:
  - Label ratio ≈ 1 words and OOV by left context

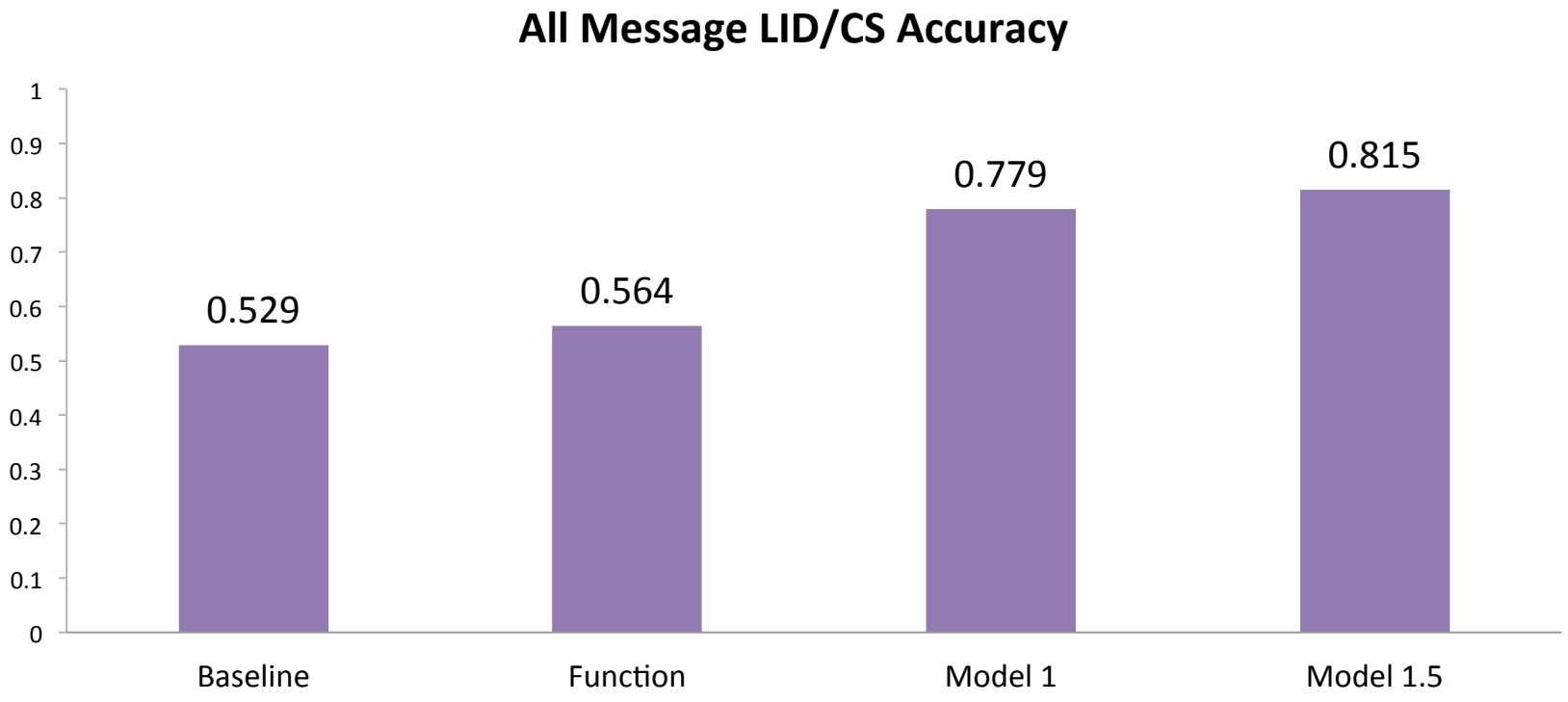
Data collection*

- Accuracy verified using test set created by bilingual Mechanical Turk annotators
  - Instructed to account for standard borrowings
- Labeled 10,000 tweets from MITRE Spanish Twitter (N = 93,136 tweets) (Burger et al. 2011) for source language
  - Token inter-annotator agreement: 96.52%
  - Tokens labeled with basic entity tags (name, title) and excluded from token evaluation

*Many thanks to Jacob Lozano Ramirez for implementing the HIT interface and performing adjudication.

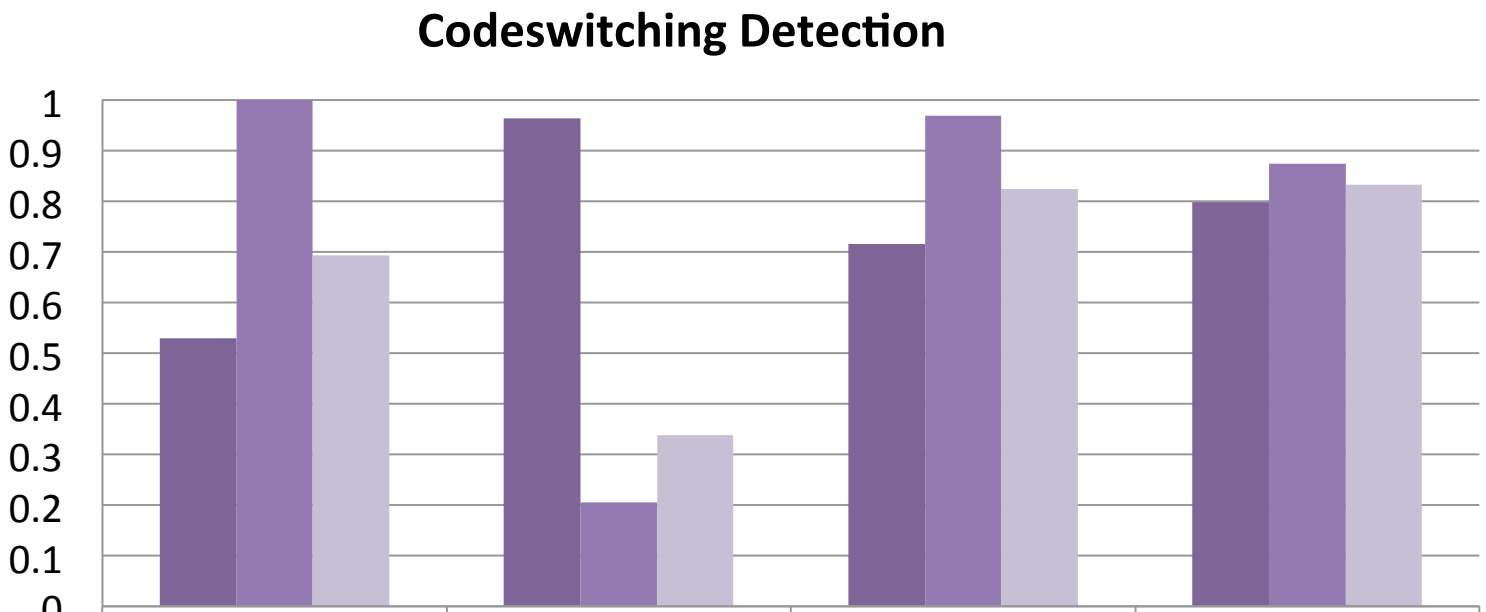
LID and Codeswitching accuracy

- Task: Mark each tweet (N=7,018) as Spanish, English, or Codeswitched
- Application: split data into single or mixed language sets



Codeswitching detection

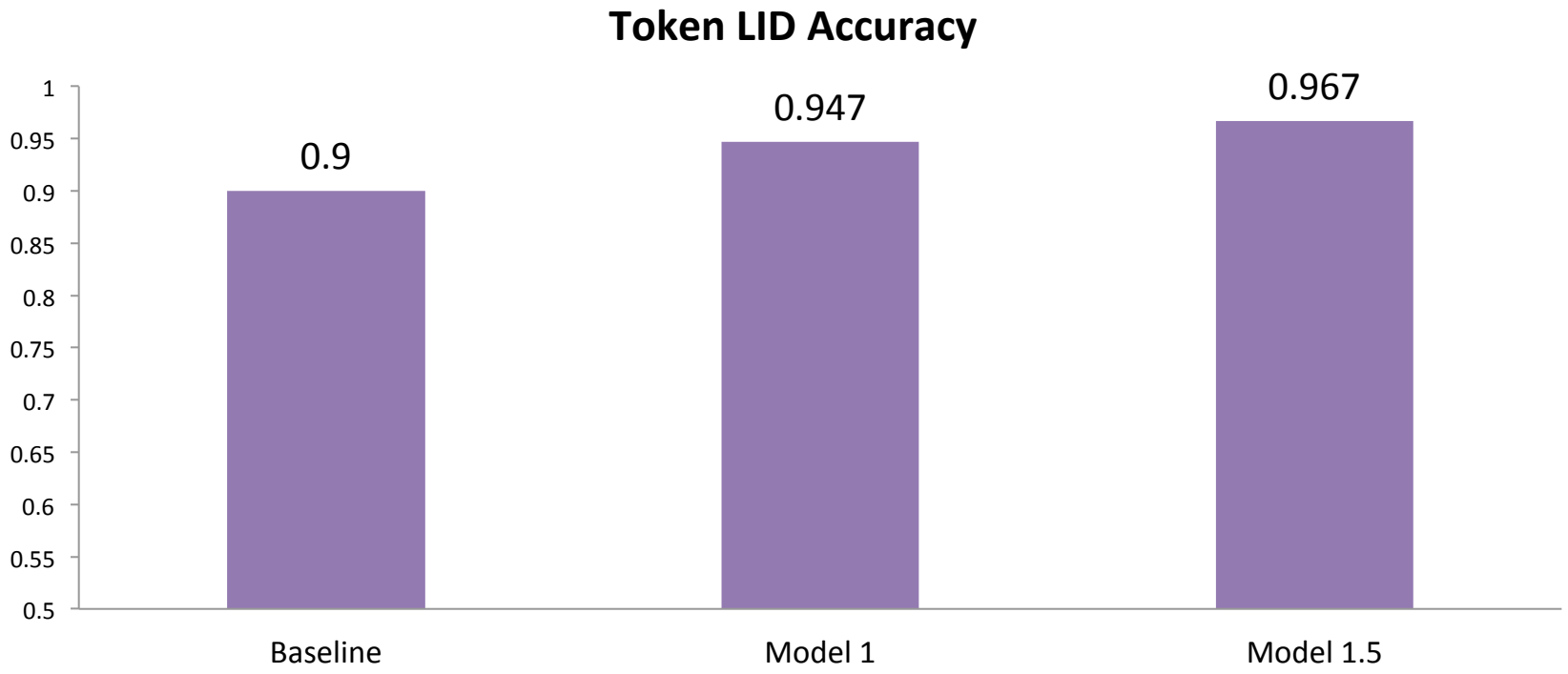
- Task: Mark each tweet (N=7,018) as Codeswitched or not
- Application: identify codeswitched tweets/users



	Baseline	Function	Model 1	Model 1.5
CS Prec.	0.529	0.963	0.715	0.797
CS Rec.	1	0.205	0.969	0.874
CS F1	0.692	0.338	0.823	0.833

Token language identification

- Task: Among codeswitched tweets (N=3,711), identify source language of each token
  - Inter-annotator agreement: 96.52%
- Application: identify tokens/spans for normalization/MT



Extensions

- Applications in linguistic research: what are the constraints on codeswitching in large-scale data? (ask me)
- Could apply more powerful learning methods:
  - Simple one state per language HMM: equivalent performance
  - SVM-HMM with morphological features: equivalent performance
  - Higher overfitting risk with these models
- Most improvement is likely to come from better OOV modeling
- Comparison with standard LID systems
  - Not yet fair as they are meant to do many-way LID, not two-way

Codeswitching: Conclusions

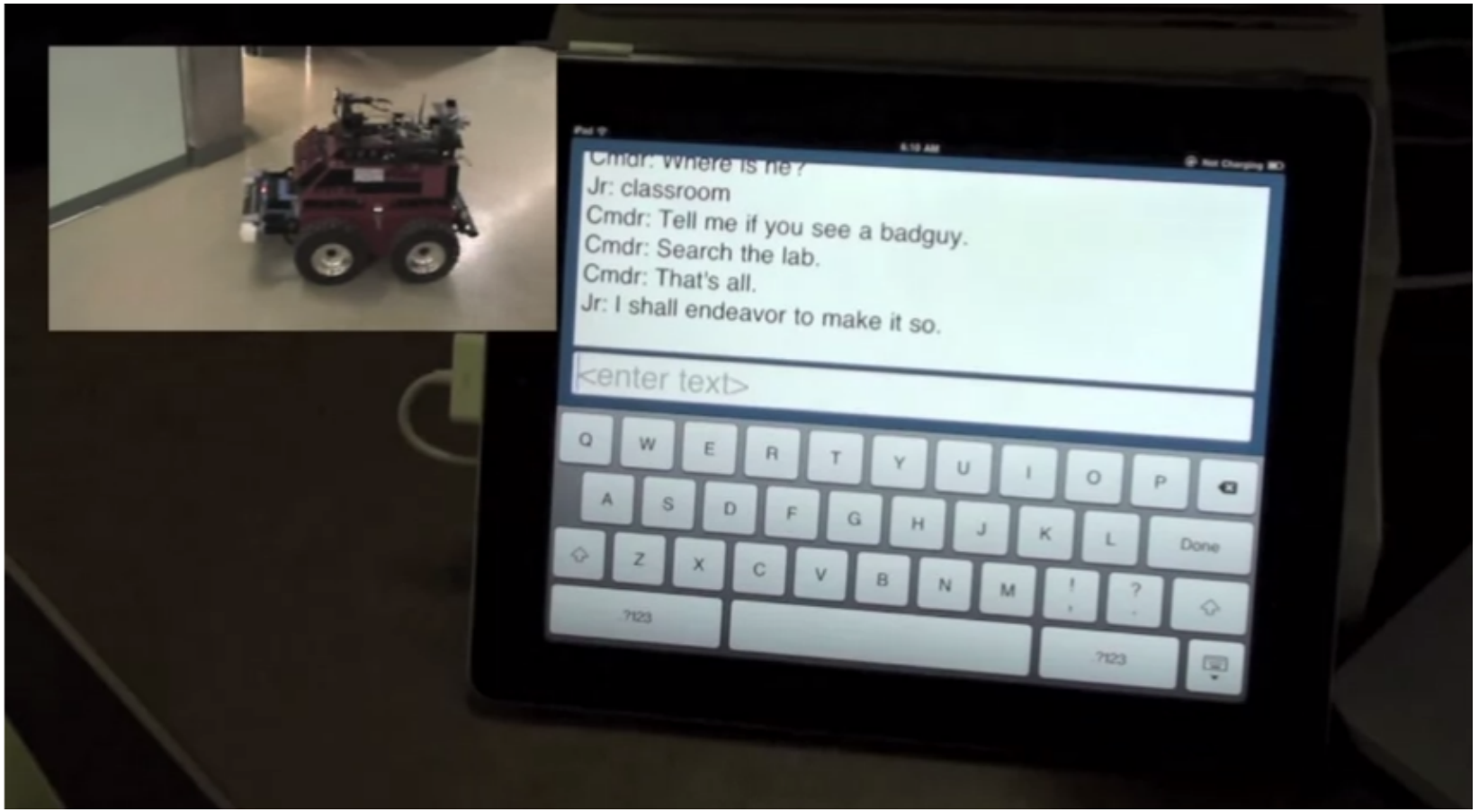
- Simple approach that takes advantage of linguistic constraints works as well as human annotators
- Fast, deterministic decoding and minimal storage (1 bit/ word)
- Promising future for improving text normalization, communicant characterization, and machine translation

IV. Conclusions

Care for some more insane heuristic?

Yes, please. “Necessity is the mother of invention,” and considering additional constraints (e.g., fast, cognitively plausible) has lead to good results.

Natural language for robot control

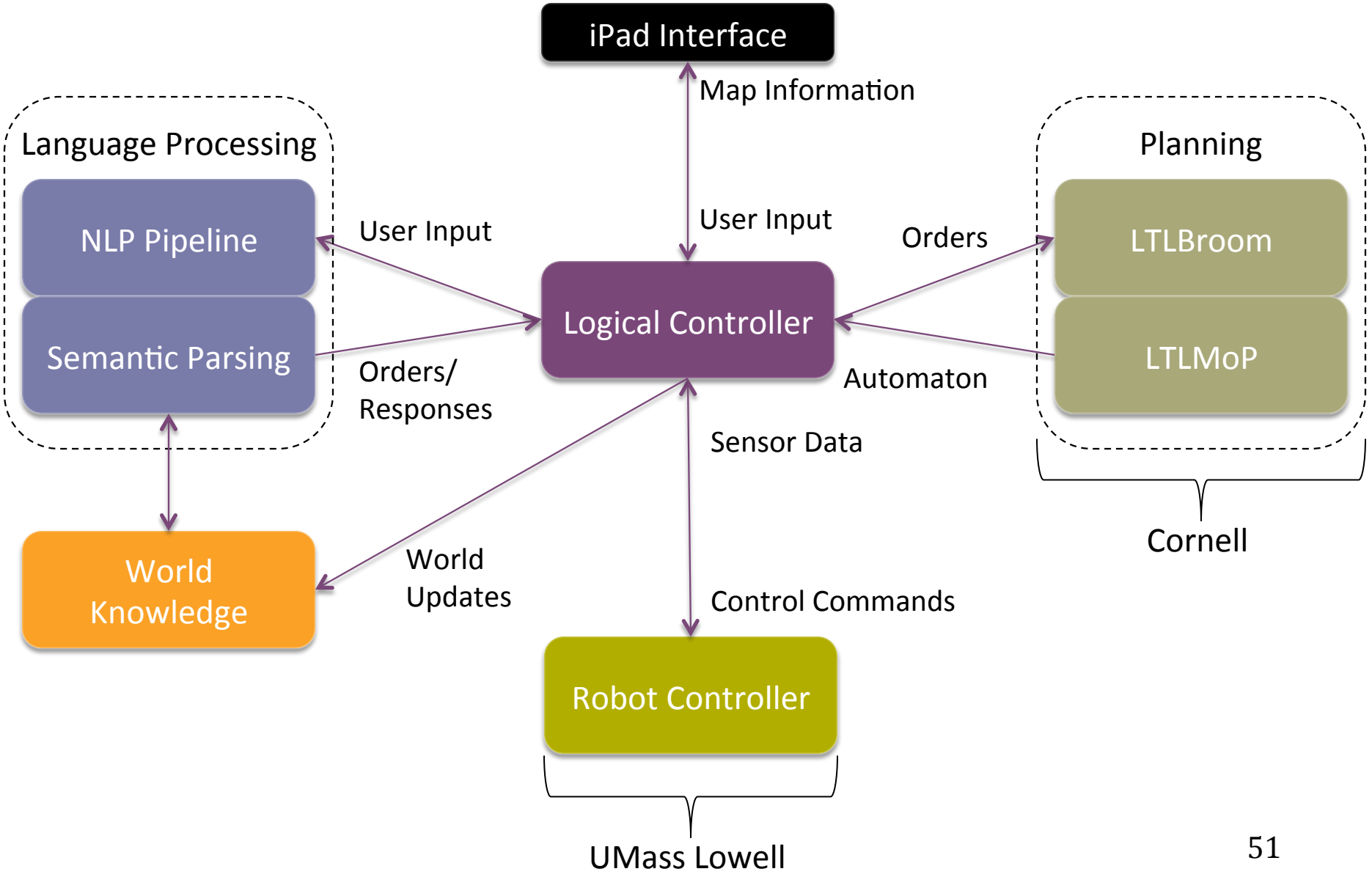


Natural language used to generate linear temporal logic plans (AAAI Language Grounding Workshop 2012; HRI 2012)

https://github.com/PennNLP/SLURP

(Supported by ARO MURI)

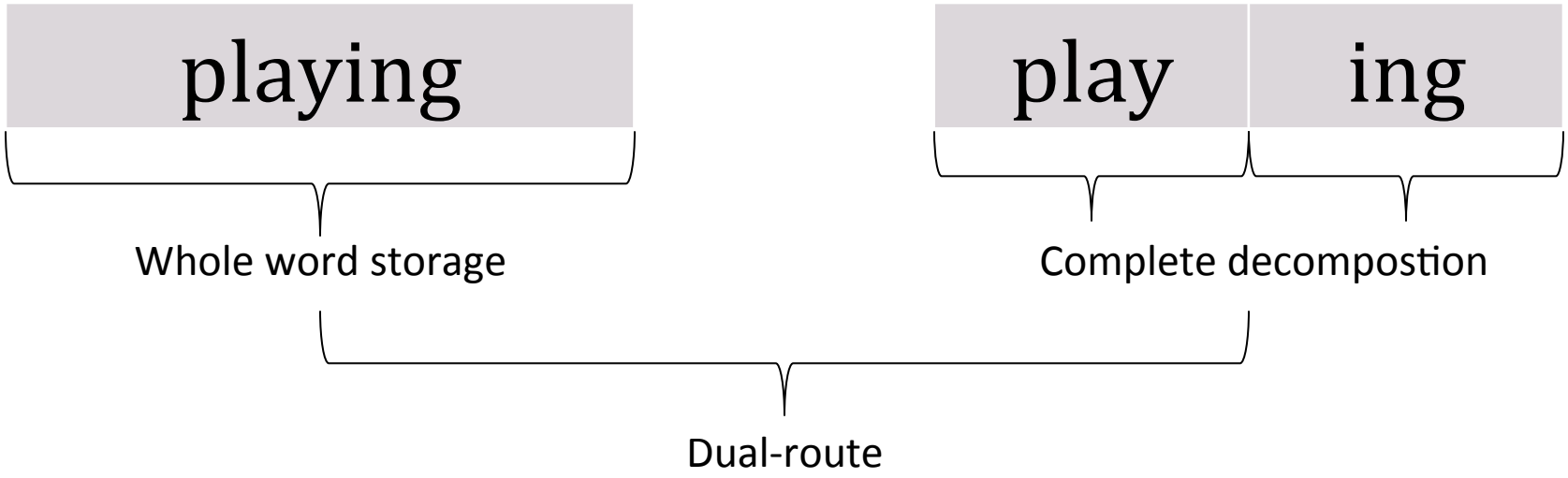
Natural language for robot control



Morphological processing

How do we represent and process morphologically complex forms? Does frequency affect our answer?

Use big data to compare whole word, decompositional, and dual-route models



Developing algorithmic-level model of morphological processing

(Chicago Linguistic Society 2012)

Conclusions

- Combining the best of statistical learning and symbolic constraints leads to good results and *fast* systems
- Cognitive/linguistic constraints provide robust and *a priori* motivated restrictions on language systems

Thanks to:
Charles Yang
Mitch Marcus
NSF IGERT #50504487
ARO MURI (SUBTLE) W911NF-07-1-0216

Constantine Lignos
lignos@cis.upenn.edu
http://www.seas.upenn.edu/~lignos