

TTIC 31210:
Advanced Natural Language Processing

Kevin Gimpel
Spring 2019

Lecture 16:
Finish Bayesian Nonparametrics;
Research Advice

Roadmap

- intro (1 lecture)
- deep learning for NLP (5 lectures)
- structured prediction (4.5 lectures)
- generative models, latent variables, unsupervised learning, variational autoencoders (1.5 lectures)
- Bayesian methods in NLP (2 lectures)
- **Bayesian nonparametrics in NLP (1.5 lectures)**
- research advice (0.5 lectures)

Assignments

- questions about Assignment 4?

Grading

- let me know before your grading deadline if you want to take this class pass/fail

Dirichlet Process (DP)

- “distribution over distributions”
- unlike Dirichlet distribution, DP does not require pre-specifying number of components (“nonparametric”)

Views of the Dirichlet Process (DP)

- last week: the “stick-breaking” view
- today we’ll discuss the “Chinese Restaurant Process” view
- with both, we have the same DP hyperparameters (base distribution & concentration parameter s)

Base Distribution G_0 for DP

- our unbounded distribution over items will choose them from base distribution
- base distribution usually has infinite support
- simple example base distribution for our morph lexicon:

$$G_0(m) = p_{\text{len}}(|m|) \prod_{i=1}^{|m|} p_{\text{char}}(m_i)$$

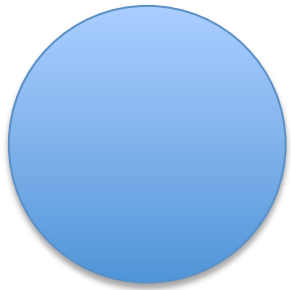
Concentration Parameter

- in stick-breaking process, concentration parameter determines how much of the stick we break off each time
- high concentration == small parts of stick

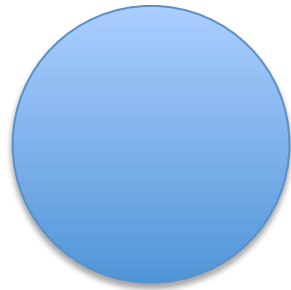


- stick-breaking construction is useful for specifying models and defining inference algorithms
- another useful way of representing a draw from a DP is with the Chinese Restaurant Process (CRP)
 - CRP provides a distribution over partitions with an unbounded number of parts

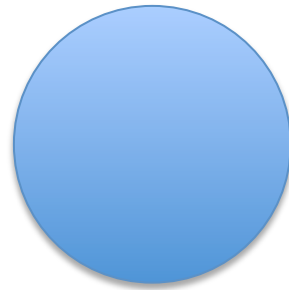
- imagine a Chinese restaurant with an infinite number of tables...



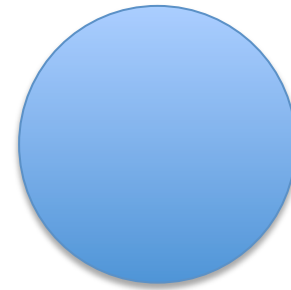
1



2



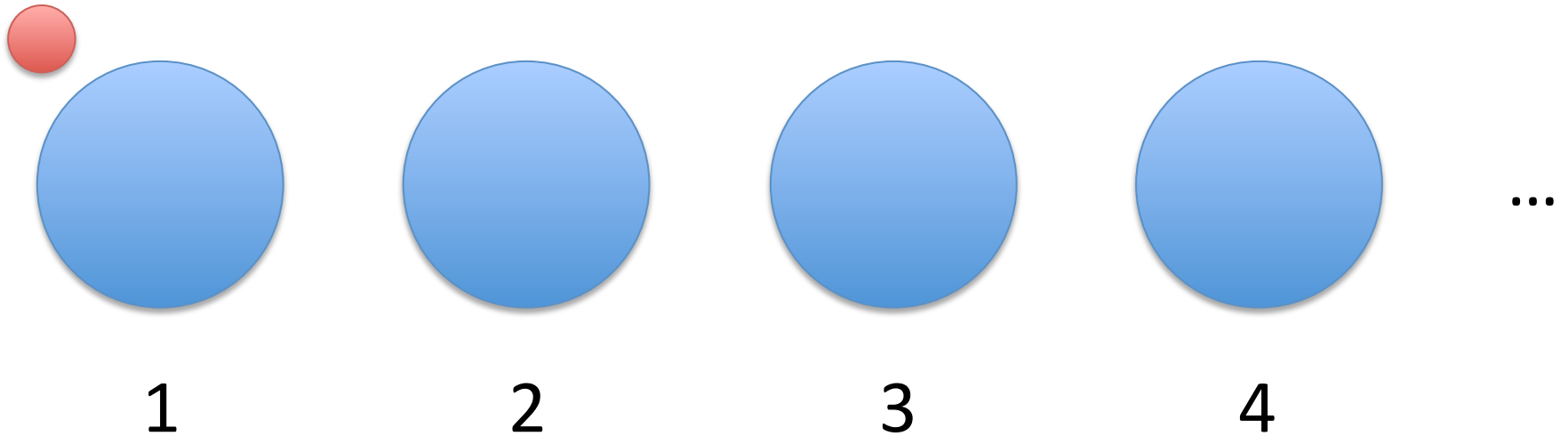
3



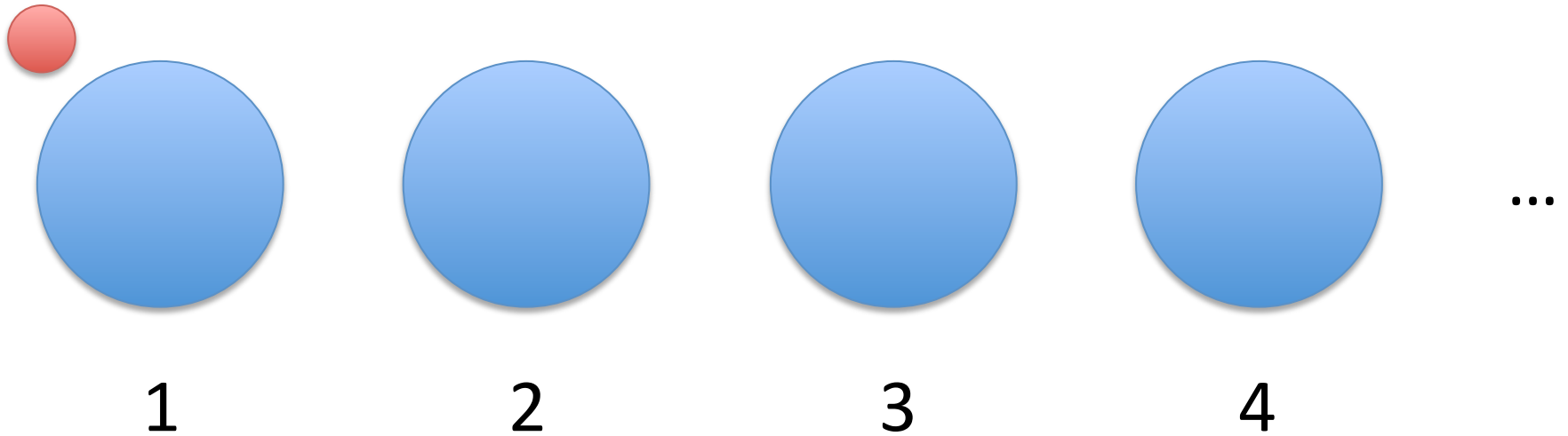
4

...

- first customer sits at first table:

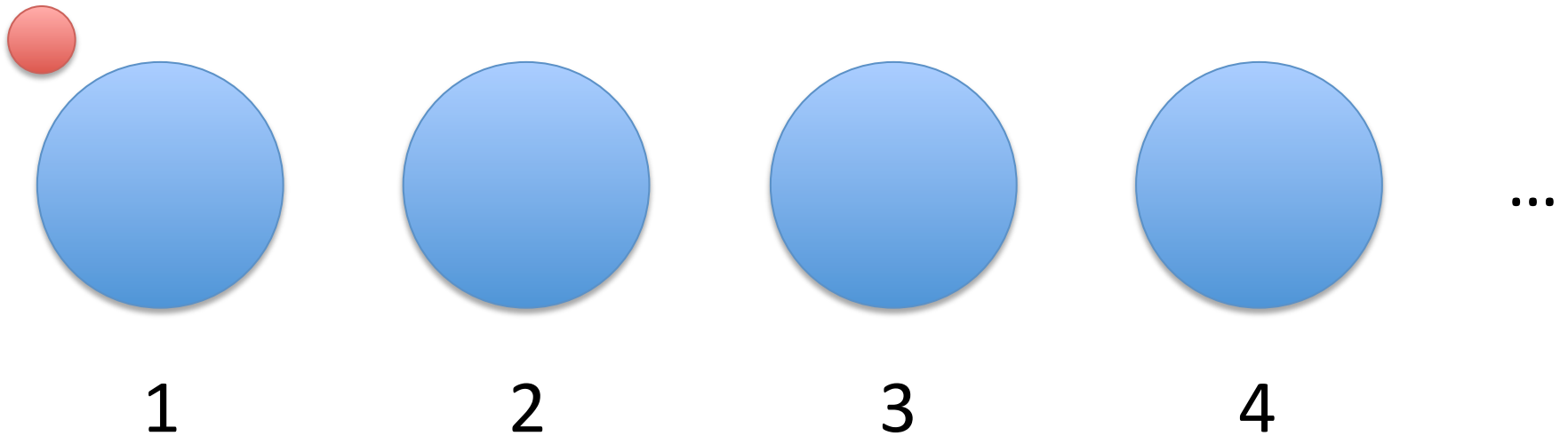


- second customer  enters, chooses a table:



- second customer  enters,

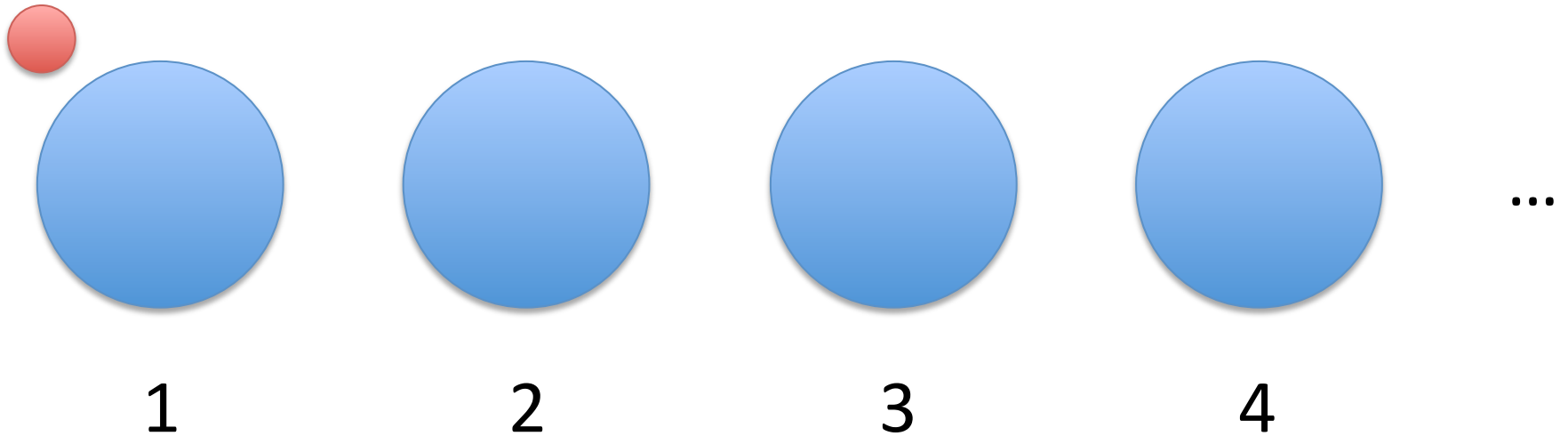
chooses table 1: $p(Y^{(2)} = 1 \mid Y^{(1)}, s) = \frac{1}{1 + s}$




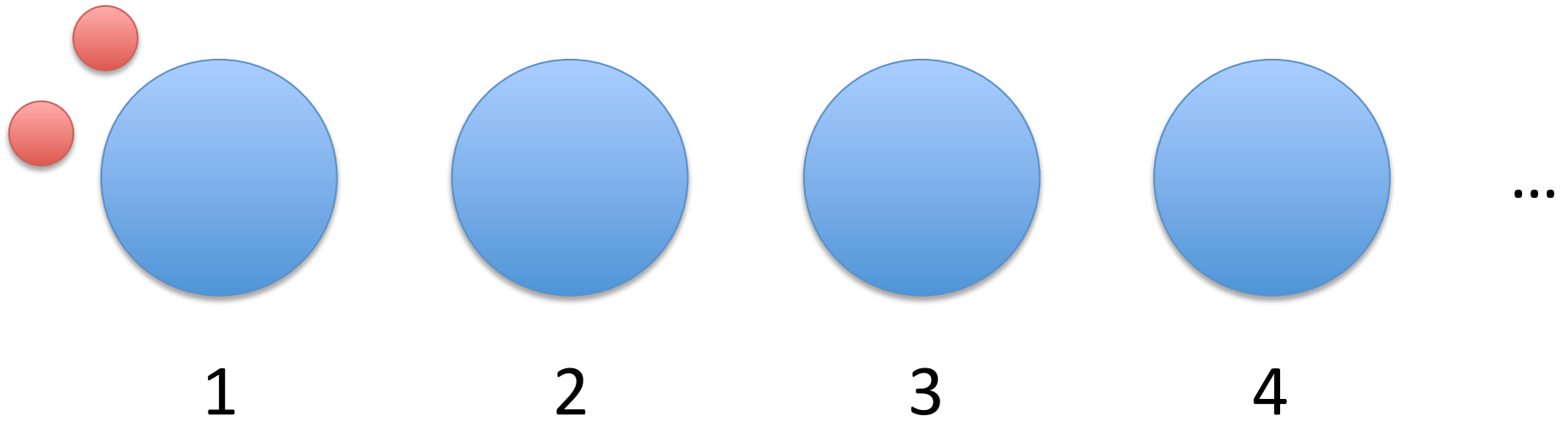
- second customer  enters,

chooses table 1: $p(Y^{(2)} = 1 \mid Y^{(1)}, s) = \frac{1}{1 + s}$

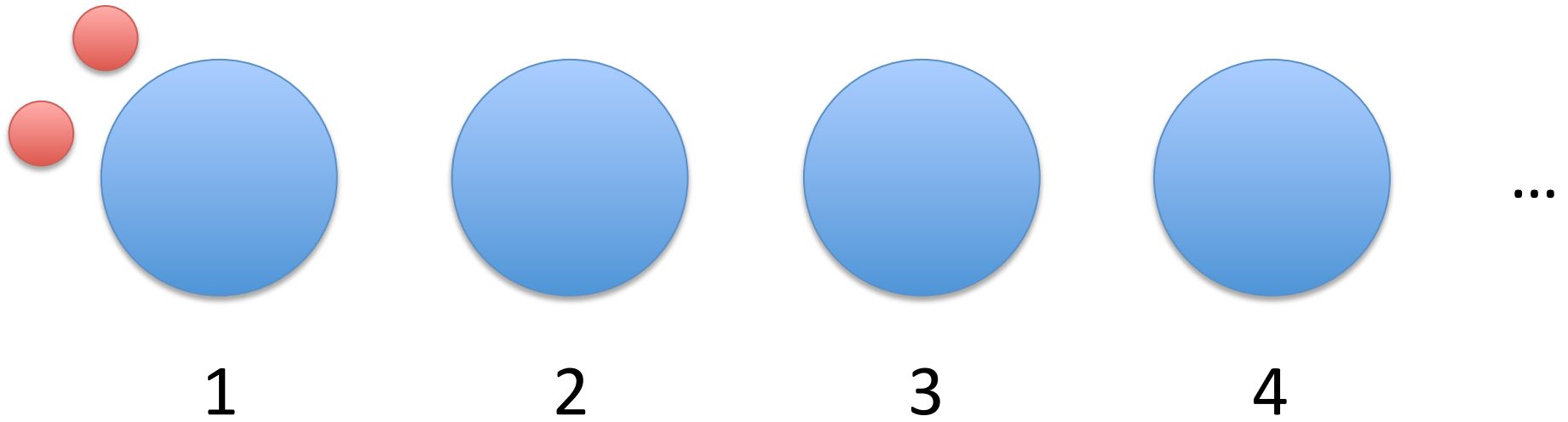
chooses new table: $p(Y^{(2)} = 2 \mid Y^{(1)}, s) = \frac{s}{1 + s}$



- second customer  enters,
chooses table 1



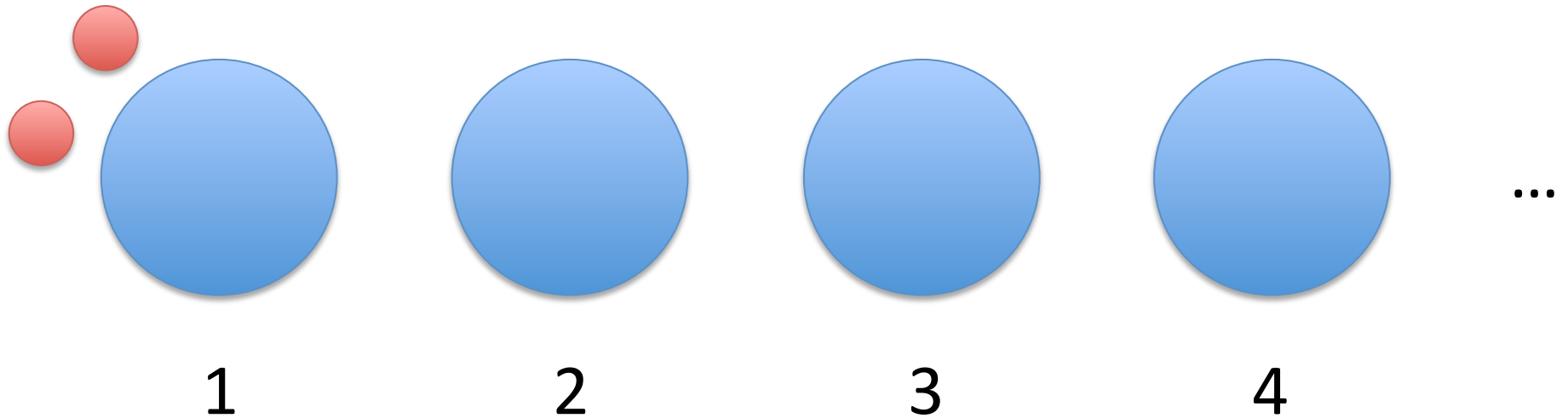
- third customer  enters,



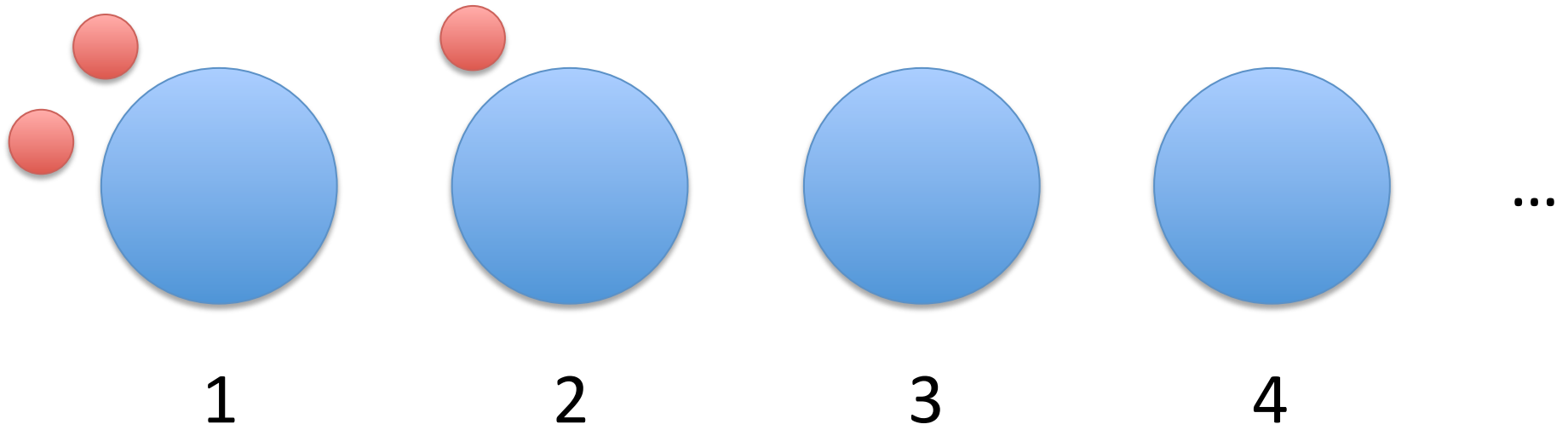
- third customer  enters,

chooses table 1: $p(Y^{(3)} = 1 \mid Y^{(1)}, Y^{(2)}, s) = \frac{2}{2 + s}$

chooses new table: $p(Y^{(3)} = 2 \mid Y^{(1)}, Y^{(2)}, s) = \frac{s}{2 + s}$



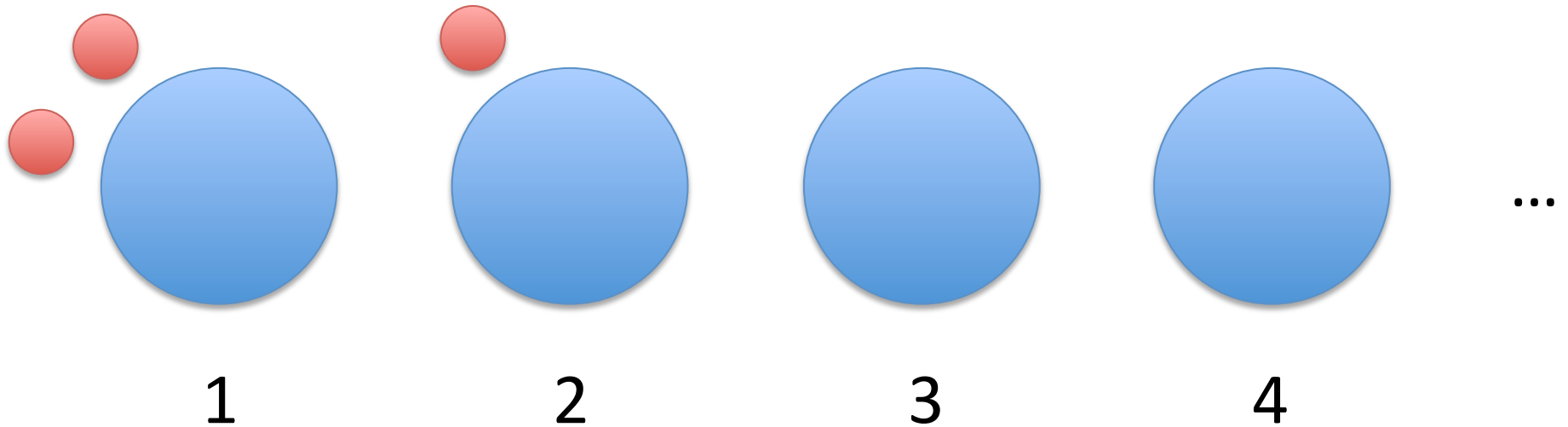
- third customer  enters,
chooses new table

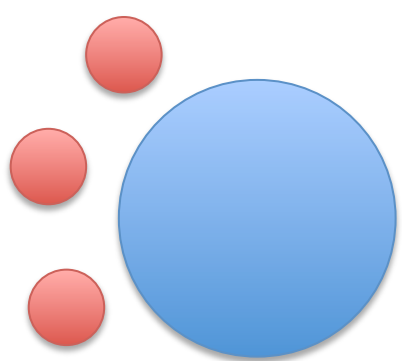


- fourth customer  enters,

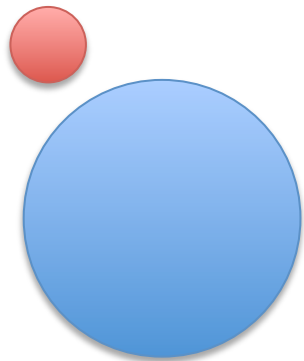
$$p(\text{choose table 1}): \frac{2}{3+s} \quad p(\text{choose table 2}): \frac{1}{3+s}$$

$$p(\text{choose new table}): \frac{s}{3+s}$$

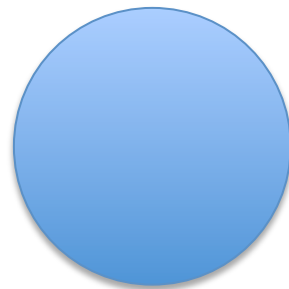




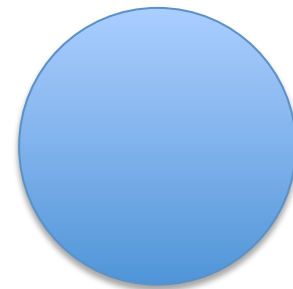
1



2



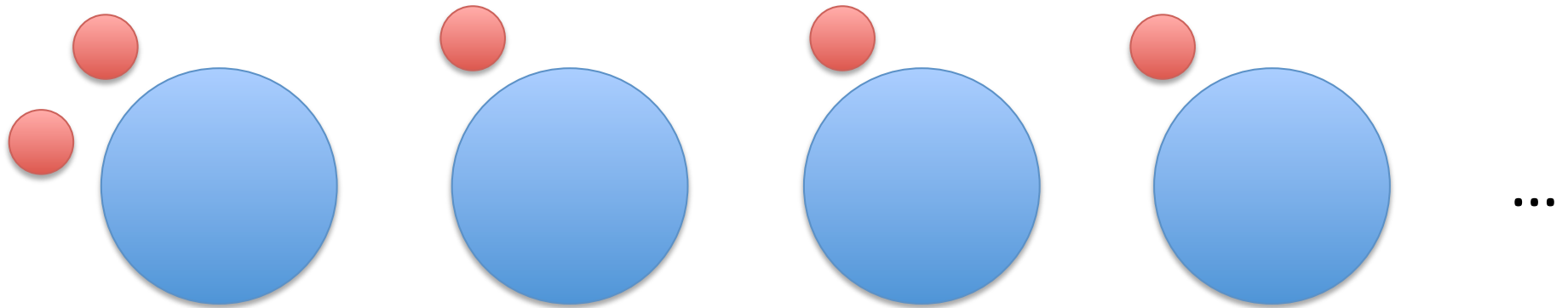
3



4

...

- large value of concentration parameter s :



1

2

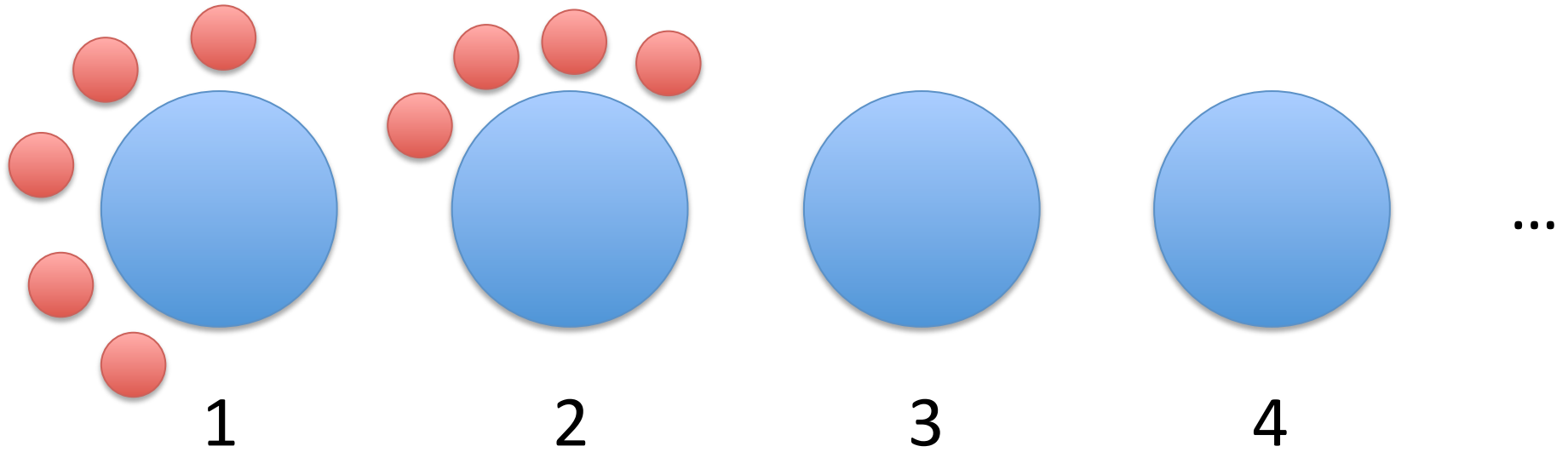
3

4

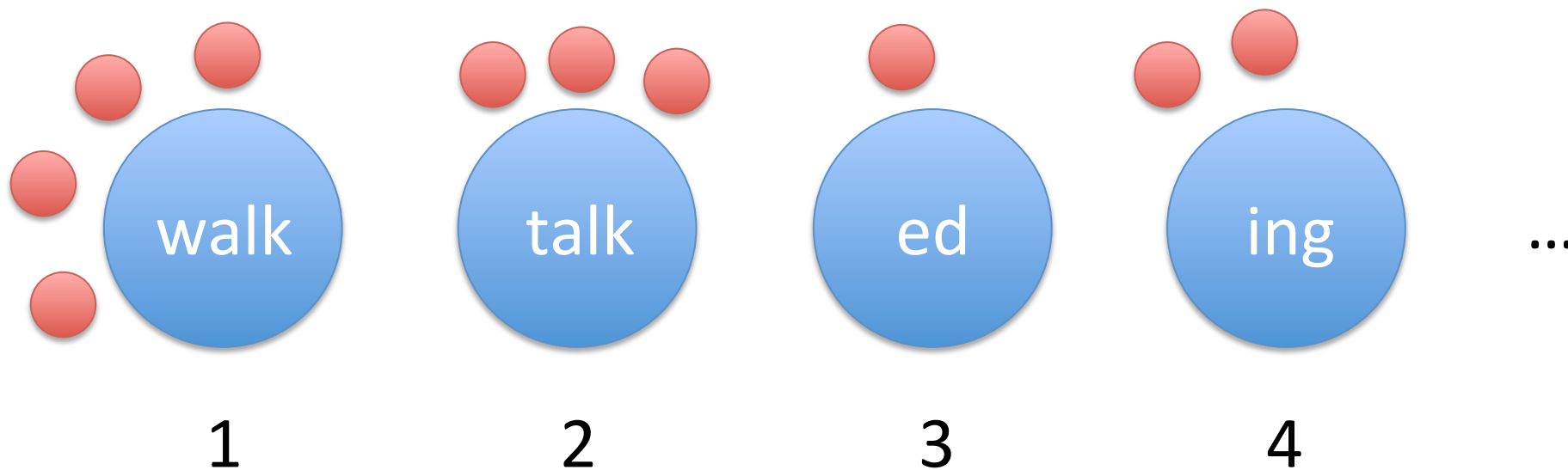
full stick





- small value of concentration parameter s :



- CRP gives us an unbounded set of probabilities
- atoms (drawn from base distribution) are the “dish” eaten by all customers at the table:





A Draw G from a DP (Stick-Breaking Representation)

- 1: $\beta \sim \text{GEM}(s)$  draw infinite probabilities from stick-breaking process with parameter s
- 2: $\theta_1, \theta_2, \dots \sim G_0$  draw atoms from base distribution
atoms can be repeated!
- 3: the distribution G is defined as:

$$G(\theta) = \sum_{k=1}^{\infty} \beta_k \mathbb{I}[\theta = \theta_k]$$

$$G(\text{"ing"}) = \sum_{k=1}^{\infty} \beta_k \mathbb{I}[\text{"ing"} = \theta_k]$$


A Representation of G Drawn from a DP (Chinese Restaurant Process Representation)

- 1: $y^{(1)}, \dots, y^{(n)} \sim \text{CRP}(s)$  draw table assignments for n customers with parameter s
- 2: $\phi_1, \dots, \phi_{y_{\max}} \sim G_0$  for each occupied table, draw atom from base distribution
- 3: set each $\theta^{(i)}$ to $\phi_{y^{(i)}}$ for $i \in \{1, \dots, n\}$

each draw from G is an atom, where its probability comes from the number of customers at its table

this avoids directly drawing G itself

$$y_{\max} = \max_i y^{(i)}$$

 number of tables occupied

- CRP can be used to define predictive distribution of values from a DP without representing G explicitly:

$$p(\theta_i \mid \boldsymbol{\theta}_{-i}, G_0, s)$$

- given that we saw values of some data points, what's the probability of seeing some value next?
- e.g., if using a DP prior for modeling words, this is the probability of the next word conditioned on all previous words

- predictive distribution of values:

$$p(\theta_i \mid \boldsymbol{\theta}_{-i}, G_0, s)$$

- integrate out G :

$$p(\theta_i \mid \boldsymbol{\theta}_{-i}, G_0, s) = \int p(\theta_i \mid G) p(G \mid \boldsymbol{\theta}_{-i}, G_0, s) dG$$

- under certain conditions (models typically used in NLP), this yields the CRP representation:

$$p(\theta_i = \theta \mid \boldsymbol{\theta}_{-i}, G_0, s) = \frac{1}{i - 1 + s} \sum_{j=1}^{i-1} \mathbb{I}[\theta_j = \theta] + \frac{s}{i - 1 + s} G_0(\theta)$$

$$p(\theta_i = \theta \mid \boldsymbol{\theta}_{-i}, G_0, s) = \underbrace{\frac{1}{i-1+s} \sum_{j=1}^{i-1} \mathbb{I}[\theta_j = \theta]}_{\text{probability of choosing some existing table with dish } \theta} + \underbrace{\frac{s}{i-1+s} G_0(\theta)}_{\text{probability of choosing a new table and drawing dish } \theta}$$

$$p(\theta_i = \theta \mid \boldsymbol{\theta}_{-i}, G_0, s) = \frac{n_{\theta}^{(\boldsymbol{\theta}_{-i})} + sG_0(\theta)}{i-1+s}$$

$n_{\theta}^{(\boldsymbol{\theta}_{-i})}$ = number of times θ was observed in cases other than i

Inference

- CRP representation of DP very useful for MCMC algorithms!
- another important detail: DP models are **exchangeable**:
 - probability of a sequence of words doesn't depend on their ordering in the sequence

Important Extensions

- hierarchical Dirichlet processes:
 - Dirichlet process where base distribution is *another* Dirichlet process!
- Pitman-Yor processes
 - generalization of Dirichlet Process that supports power law effects
- hierarchical Pitman-Yor processes

Segmentation

(a)	yuwanttusiD6bUk	(b)	you want to see the book
	lUkD*z6b7wIThIzh&t		look there's a boy with his hat
	&nd6d0gi		and a doggie
	yuwanttulUk&tDIIs		you want to look at this
	lUk&tDIIs		look at this
	h&v6drINK		have a drink
	okenQ		okay now
	WAtsDIIs		what's this
	WAtsD&t		what's that
	WAtIzIt		what is it
	lUkk&nyutekItQt		look can you take it out
	tekItQt		take it out
	yuwantItIn		you want it in
	pUtD&tan		put that on
	D&t		that

Fig. 1. An excerpt from the beginning of the corpus used as input to Venkataraman's (2001) word segmentation system, showing (a) the actual input corpus and (b) the corresponding standard orthographic transcription. The corpus was originally prepared by Brent and Cartwright (1996) using data from Bernstein-Ratner (1987), and was also used as input to Brent's (1999) MBDP-1 system.

Goldwater et al. (2009): *A Bayesian framework for word segmentation: Exploring the effects of context*

Segmentation

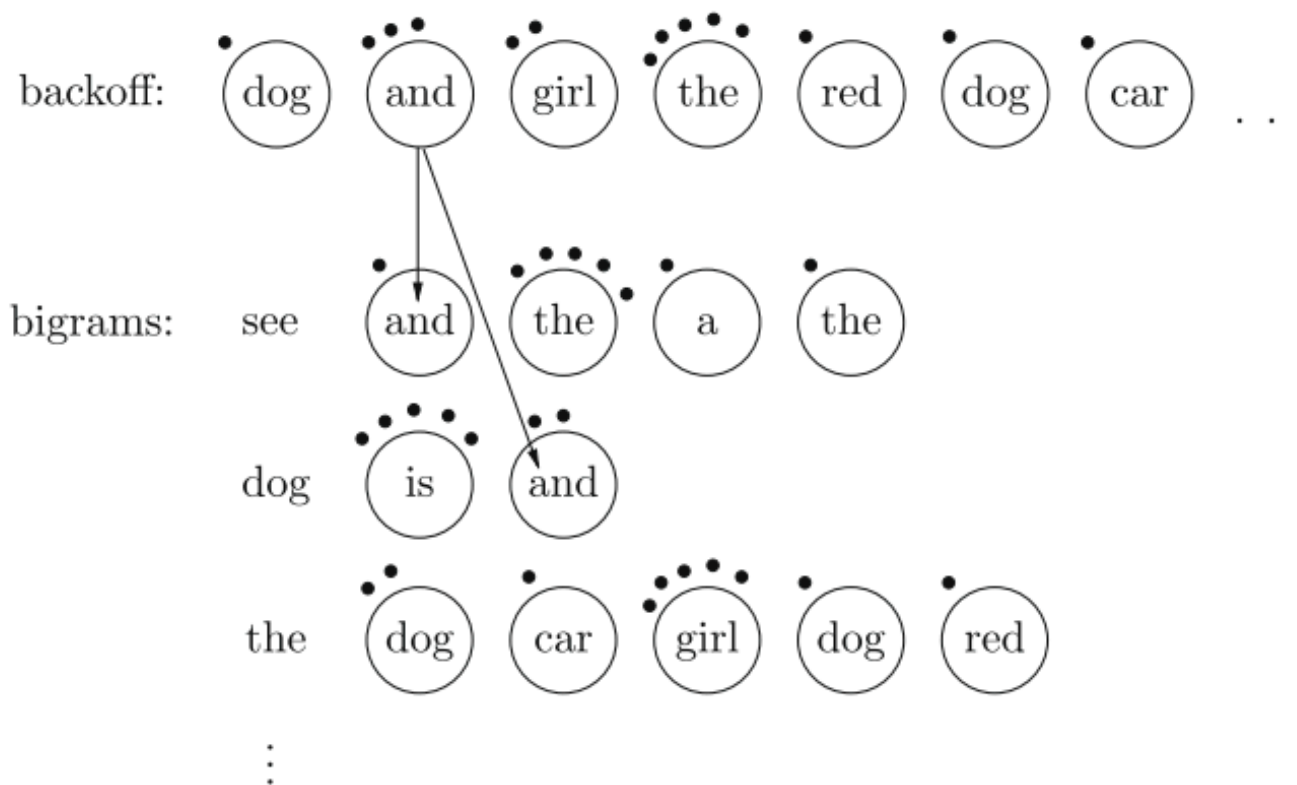


Fig. A5. Bigrams are modeled using a hierarchical Chinese restaurant process. Each lexical item ℓ has its own restaurant to represent the distribution of tokens following ℓ in the data. The labels on the tables in these bigram restaurants are drawn from the distribution in the backoff or “master” restaurant (top). Each customer (black dot) in the bigram restaurants represents a bigram token; each customer in the backoff restaurant represents a label on some bigram table.

Goldwater et al. (2009): *A Bayesian framework for word segmentation: Exploring the effects of context*

Synchronous Grammar Rules for Machine Translation

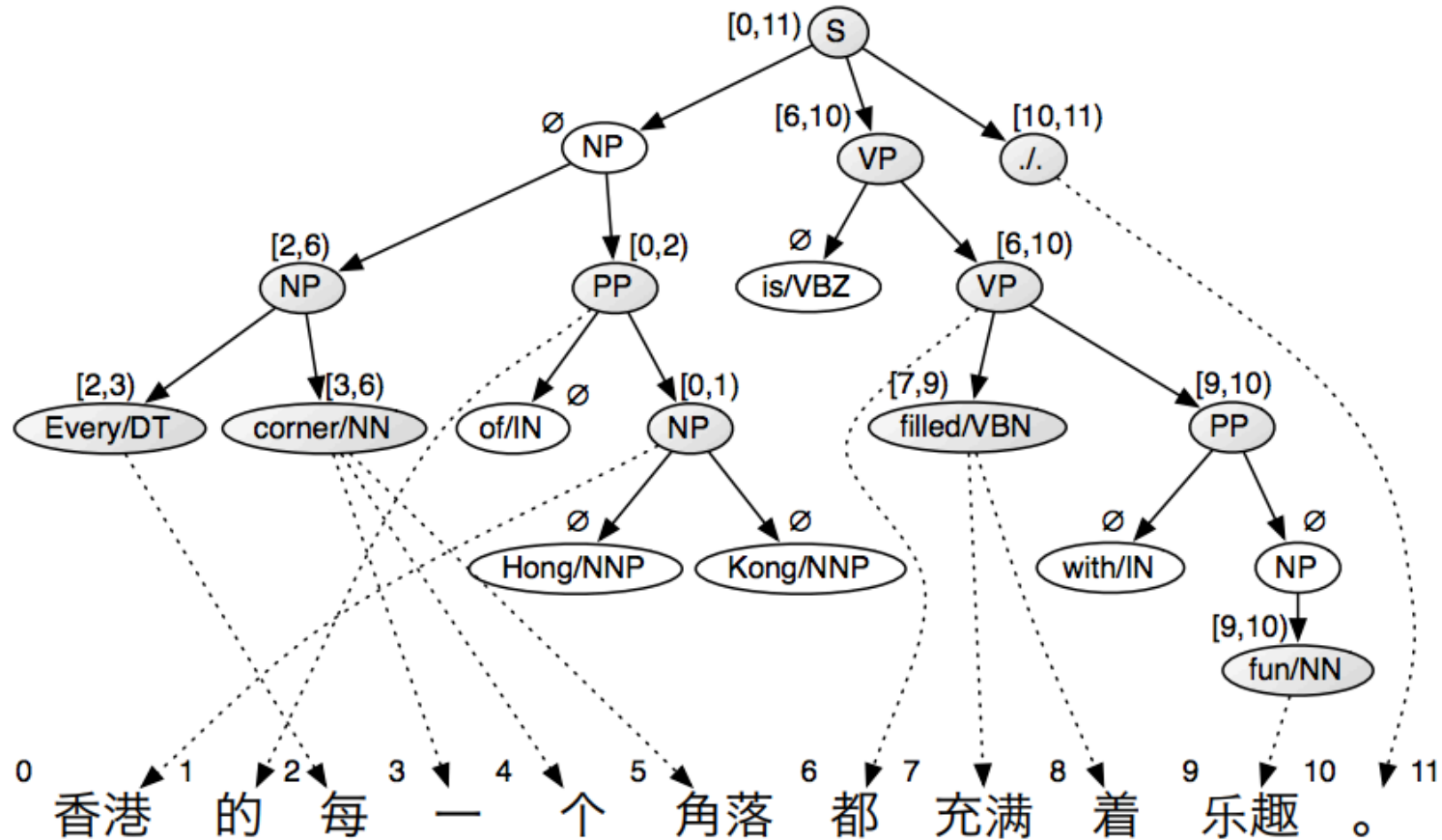
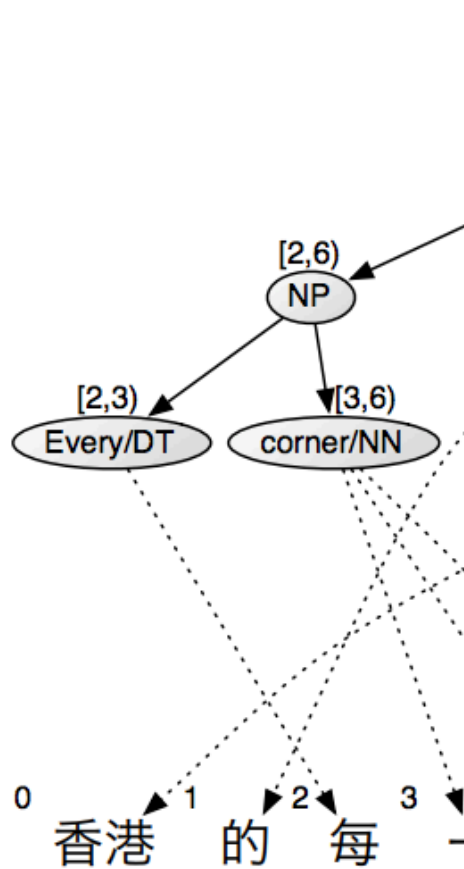


Figure 1: Example derivation. Each node is annotated with their span in the target string (aligned nodes are shaded). The dotted edges show the implied alignments. Preterminals are displayed with their child terminal in the leaf nodes.

Cohn & Blunsom (2009): *A Bayesian Model of Syntax-Directed Tree to String Grammar Induction*

Synchronous Grammar Rules for Machine Translation



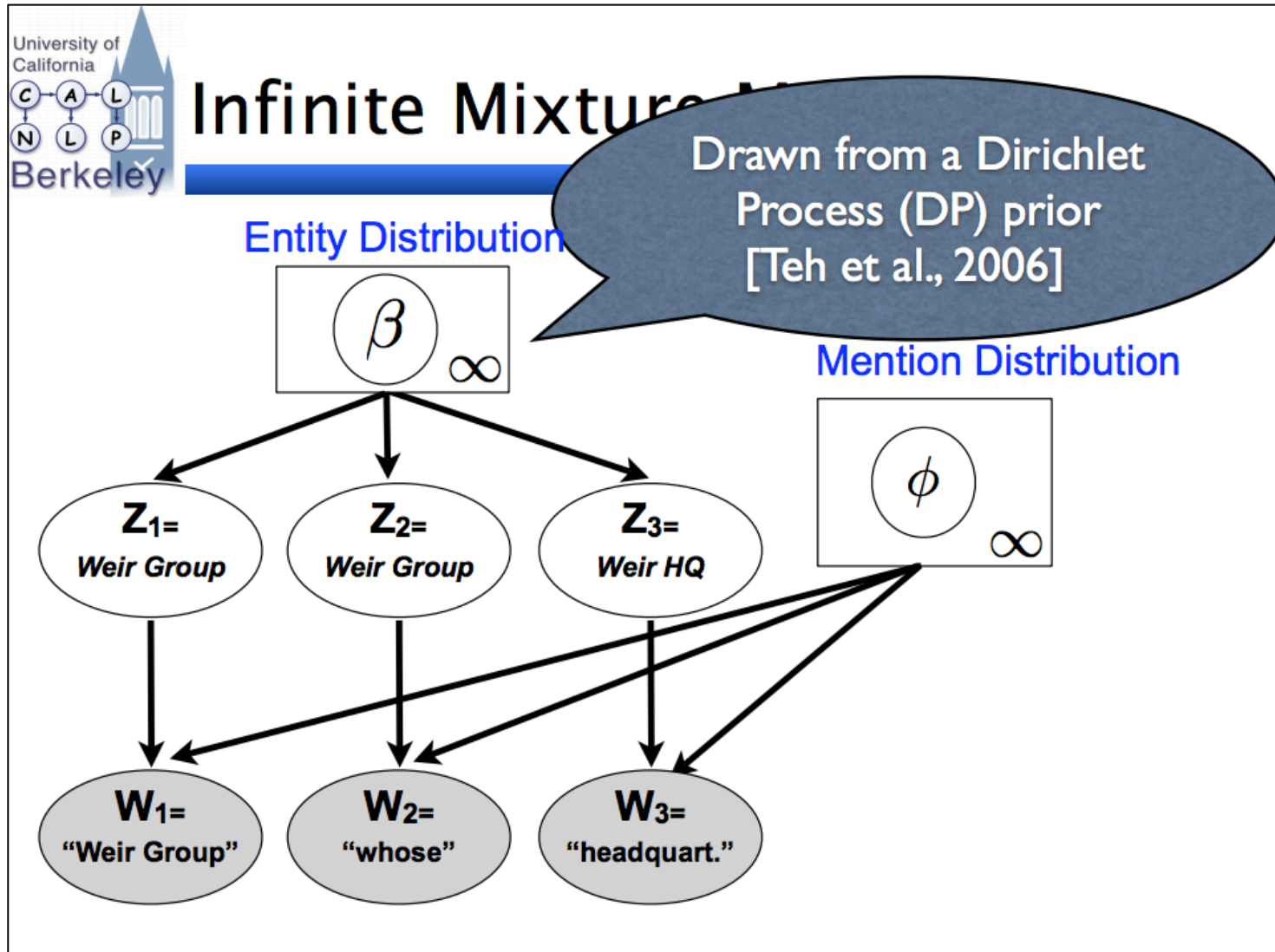
[0,11) (e)

$\langle (S (NP (NP_1 PP_2) VP_3) .4), [2] [1] [3] [4]) \rangle$
$\langle (NP DT_1 NN_2), [1] [2] \rangle$
$\langle (DT Every), 每 \rangle$
$\langle (NN corner), 一个角落 \rangle$
$\langle (PP (IN of) NP_1), [1] 的 \rangle$
$\langle (NP (NNP Hong) (NNP Kong)), 香港 \rangle$
$\langle (VP (VBZ is) VP_1), [1] \rangle$
$\langle (VP VBN_1 PP_2), 都 [1] [2] \rangle$
$\langle (VBN filled), 充满着 \rangle$
$\langle (PP (IN with) (NP NN_1)), [1] \rangle$
$\langle (NN fun), 趣 \rangle$
$\langle (. .), \circ \rangle$

Figure 1: Example derivation. Each node is a non-terminal. Solid edges show the implied alignments. Preterminal symbols are shown in italics.

Table 1: Grammar rules specified by the derivation in Figure 1. Each rule is shown as a tuple comprising a target elementary tree and a source string. Boxed numbers show the alignment between string variables and frontier non-terminals.

Unsupervised Coreference Resolution



Haghighi & Klein (2007): *Unsupervised Coreference Resolution in a Nonparametric Bayesian Model*

nato must either say " yes " or " no " to the baltic states .

- Generative story:
 - Generate number of word positions
 - Generate number of colors
 - Assign word positions to colors
 - Generate a lexical pattern for each color

nato must either say " yes " or " no " to the baltic states .

- **Generative story:**
 - Generate number of word positions ($n = 16$)
 - Generate number of colors
 - Assign word positions to colors
 - Generate a lexical pattern for each color

nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

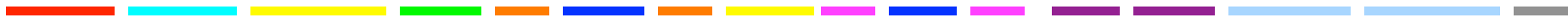
- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

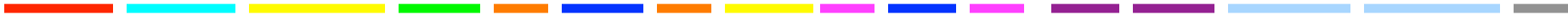
- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



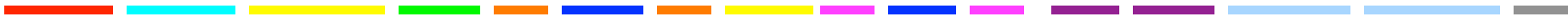
nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato



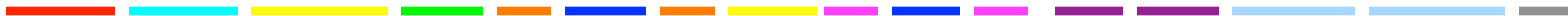
nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato must



nato must either say " yes " or " no " to the baltic states .

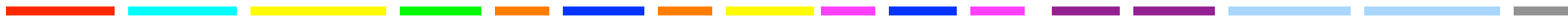
- **Generative story:**

- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato must either

or



What is a **pattern**?

A sequence of symbols, possibly including the special symbol “`__`” which is used to indicate a gap of nonzero length

Examples:

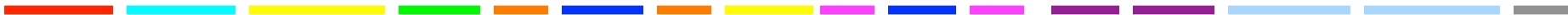
nato
must
either `__` or

the united states
according to the `__` ,
countries `__` their `__` the united states



nato must either

or



nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

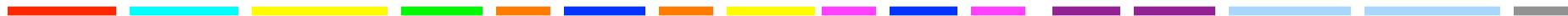
- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato must either

or

baltic states



nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color

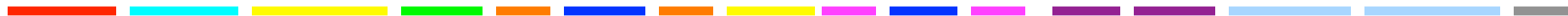


nato must either

or "

"

baltic states



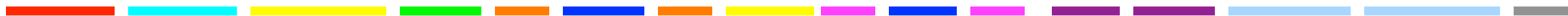
nato must either say " yes " or " no " to the baltic states .

- **Generative story:**

- Generate number of word positions ($n = 16$)
- Generate number of colors ($m = 10$)
- Assign word positions to colors
- Generate a lexical pattern for each color



nato must either say " yes " or " no " to the baltic states .



Highest-Ranked Patterns

-- __ --	how __ ?	we __ our	his __ his
(__)	the __ (__)	over __ past	some __ others
- __ -	on __ basis	prevent __ from	may __ be
both __ and	less __ than	in __ way	as __ as
not only __ but	on __ other hand	one __ another	oil __ gas
" __ "	at __ level	political __ economic	at __ moment
more __ than	it is __ that	for __ reasons	such as __ and
either __ or	not __ , but	at __ time	question __ whether
why __ ?	play __ role	more __ more	if __ then
neither __ nor	france __ germany	the rest __ world	war __ iraq
what __ ?	he __ his	more __ less	; __ ;
rule __ law	allow __ to	in __ region	have __ been
whether __ or	for __ first time	rich __ poor	in __ cases
around __ world	china __ india	as __ whole	war __ terror
has __ been	what __ do	on __ scale	at __ cost

When to be Bayesian?

- if you're doing unsupervised learning or learning with latent variables
- if you want to marginalize out some model parameters
- if you want to learn the structure/architecture of your model
- if you want to learn a potentially-unbounded lexicon or set of latent items (Bayesian nonparametrics)

Being Bayesian in Practice

- you can forgo theory to solve a problem, but you should know what you are doing!
- e.g., some researchers would start by deriving a Gibbs sampler without thinking about the model
 - as a result, the Gibbs sampler was not a Gibbs sampler (just some code that did something)
 - if you only care about solving a problem, this may be fine, but you should make sure you understand and communicate to readers that it's not a Gibbs sampler!

Roadmap

- intro (1 lecture)
- deep learning for NLP (5 lectures)
- structured prediction (4.5 lectures)
- generative models, latent variables, unsupervised learning, variational autoencoders (1.5 lectures)
- Bayesian methods in NLP (2 lectures)
- Bayesian nonparametrics in NLP (1.5 lectures)
- **research advice (0.5 lectures)**

- getting started
- researching
- communication

Getting Started on a New Project

what are the inputs and outputs?

when I talk to people who “want to use AI”:

- they want AI to tell them what to do with their data
- they have never thought about inputs and outputs

If you are the AI consultant...

ask for examples of inputs and outputs

if they don't have these, then tell them to study the data until they figure this out

most important thing they can do: think long and hard about this part, keeping in mind privacy, fairness, and ethics

Once you have input/output pairs...

look at the data

- use UNIX command-line tools: awk, grep, sort, uniq, cut, join, comm, etc.
- do what you can without writing code
- your code is slow and buggy

Once you have input/output pairs...

“If you’re going to ask the computer to do the task, try to do it yourself first.” –Kevin Knight

sample some data, hide the outputs from yourself, and try to predict them

if you find yourself using tricks and shallow heuristics, that’s what the computer will do too!

Preprocessing

be aware of the preprocessing you're doing!

I preprocessed the data for the assignments

– some of you did extra preprocessing. why?

no single preprocessing pipeline is optimal

everything can be tuned, including preprocessing

Most Important Preprocessing Rule

if using embeddings or linguistic analyzers, match your preprocessing to what they expect

this may be non-trivial – you might need to reverse-engineer their decisions

- getting started
- researching
- communication

Keep Your Goals in Mind

higher numbers are neither necessary nor sufficient for making a contribution

junior researchers want to get higher numbers

senior researchers want to know *why* the numbers are higher

“I always tell students not to get too pleased when they get state of the art performance on some standard task: someone else will beat them next year.

If the only thing that I learn from their papers is that they win on task X, then next year there's nothing to learn from that paper.

The paper has to teach me something else to have any sort of lasting effect: what is the generalizable knowledge.”

– Hal Daume, *The myth of a strong baseline*, Nov 15, 2014

Working on a Research Project

throughout, be both **quantitative** *and* **qualitative**

if **quantitative** eval is not working out, it can be hard to figure out why (it's just numbers...)

but whether or not you're beating the baseline, **qualitative** eval should look reasonable

if **qualitative** eval looks bad, maybe there's a bug

Working on a Research Project

even worse, **quantitative** eval may look good but there could be a beneficial bug

e.g., bug in evaluation code

your brain is good at sanity checks in **qualitative** evaluation

computers are not (yet) good at sanity checks

The Role of Your Brain in Your Research

use your brain to help you

- this is non-trivial; see “The Bitter Lesson” by Rich Sutton

for some things, your brain is better than a computer

for other things, a computer is better than your brain

figure out which is which

- e.g., look at data, form hypotheses, write code to check those hypotheses on more data that you can't look at systematically, repeat

you can't be systematic about everything

do a greedy search (or beam search) through the space of ideas, experiments, methods

brains are pretty good at this sort of open-ended, heuristic search

The Awkward Stage



The Final Painting



Credit: Lee Hammond

The Awkward Stage



beginning painters often
give up when they reach
the awkward stage

experienced painters
know how to finish

Credit: Lee Hammond

Research Projects have Awkward Stages

you are bogged down in hyperparameter tuning

you have tried so many random things that you don't know how to distill them into a coherent story

the results are inconsistent, helping on some tasks/datasets but not others (this is very common!)

you've been trying stuff forever and nothing seems to predictably affect the results

Escaping the Awkward Stage

embrace the notion that a long search over the space of ideas is necessary for understanding!

figure out what matters and what doesn't

talk to other researchers and see what parts of your work they are most excited about

focus on those parts, streamlining the rest

When Things Don't Work Out

sometimes you can keep working and publish a paper about your experiences

but you might be better off just leaving things behind and starting something new

publishing has costs: time and attention

- getting started
- researching
- communication

“Your writing is not about you. It’s about the Reader...
There is one thing all Readers want:
clear,
concise,
comprehensible sentences
that mean something to *them*.”

June Casagrande

It Was the Best of Sentences, It was the Worst of Sentences

Technical Writing

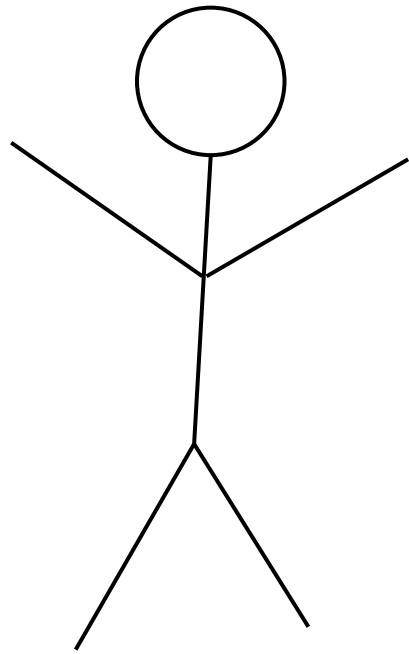
writing should be in service of the research

good writing does not draw attention to itself

mathematical notation?

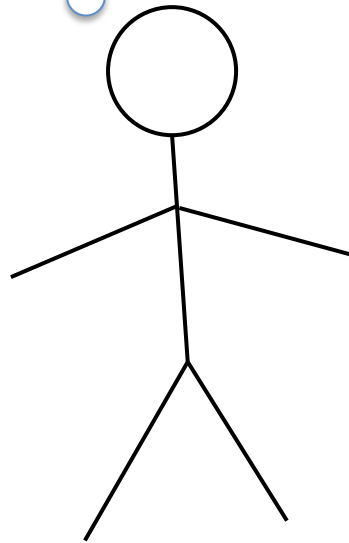
- goal is to make things clearer, not impress anyone!
- with good notation, complex ideas can be clear & concise

I hope they think I'm smart

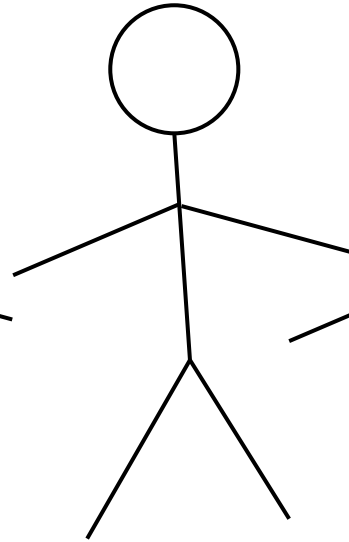


Speaker

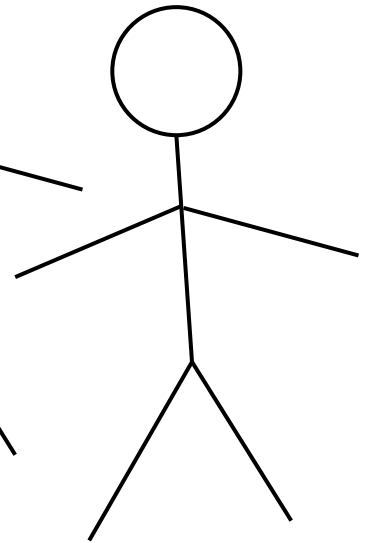
how does this relate to my research?



how does this fit into my grand unified theory?

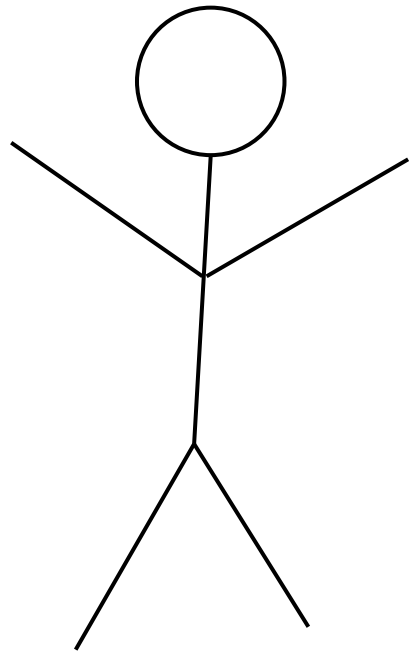


I didn't catch that technical part and now I'm lost



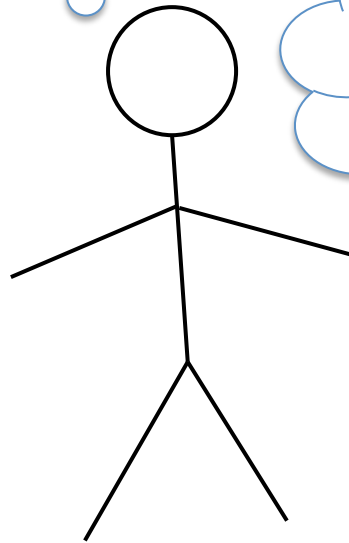
Audience

I hope they think I'm smart



Speaker

how does this relate to my research?



I'm not following. Is everyone else smarter than me?

I'm bored

Audience

how does this fit into my grand unified theory?

I didn't catch that technical part and now I'm lost

I hope they think I'm smart

how does this relate to my

how does this fit into my grand unified theory?

I didn't catch that technical part and now I'm lost

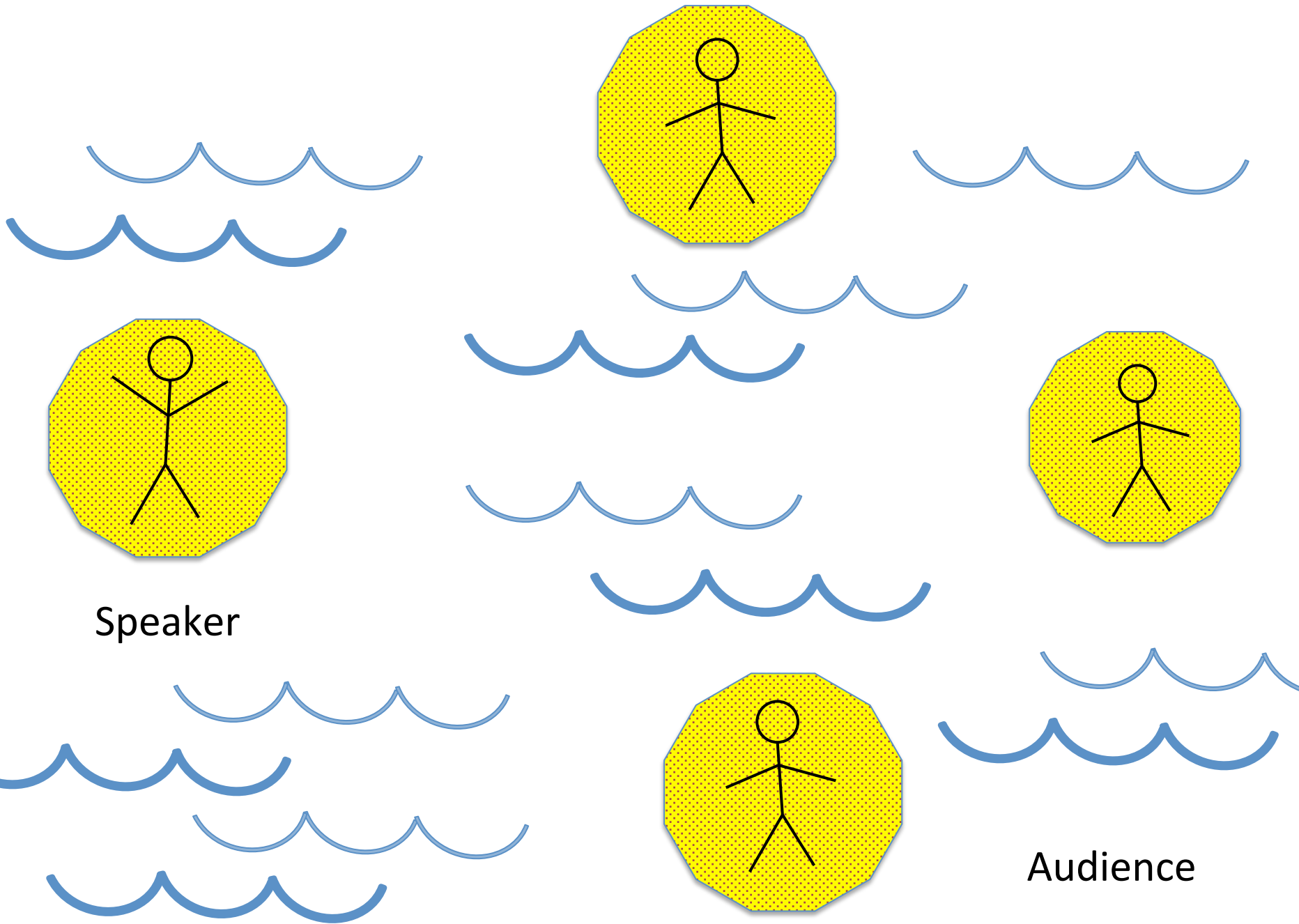
most people think about themselves most of the time

Smarter than me.

I'm bored

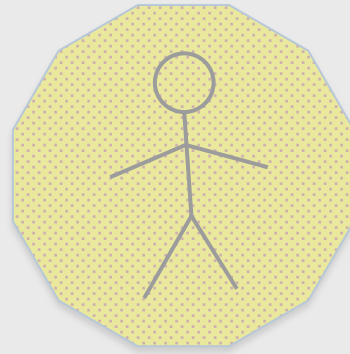
Speaker

Audience



Speaker

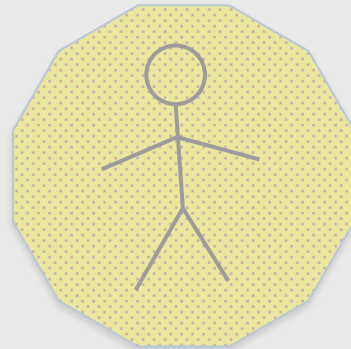
Audience



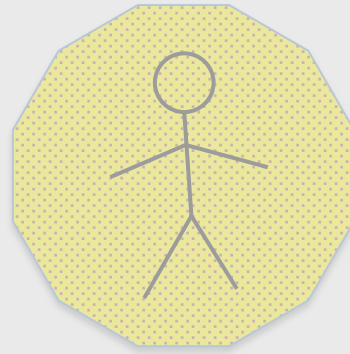
every talk is about bridging a gap



Speaker



Audience

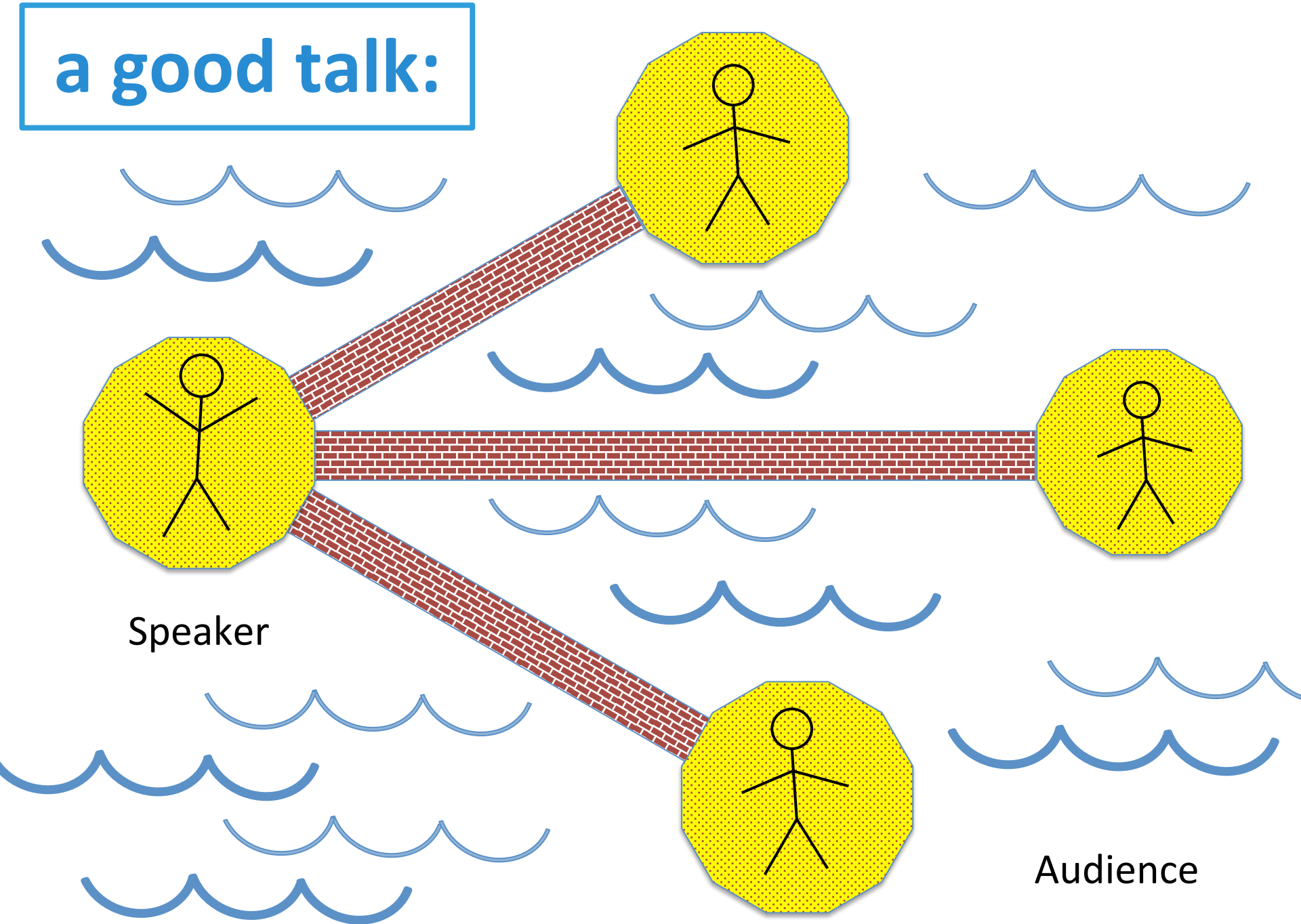


every talk is about bridging a gap

**you have something in your head,
you want to get it into their heads**

Audience

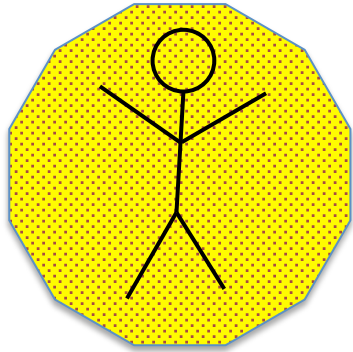
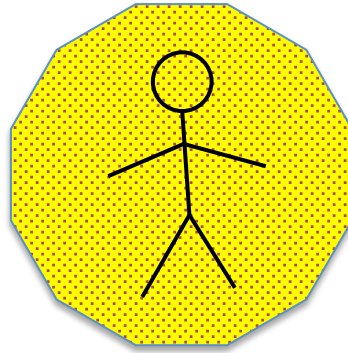
a good talk:



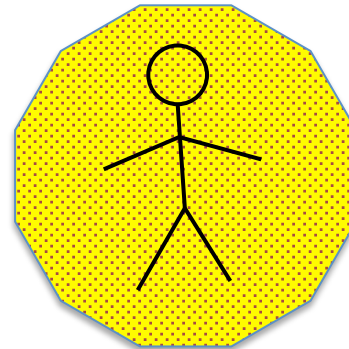
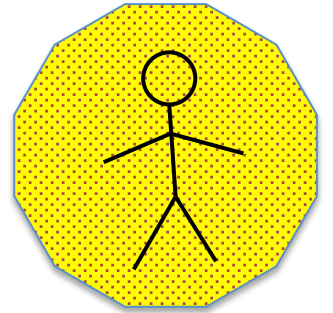
Speaker

Audience

a bad talk:

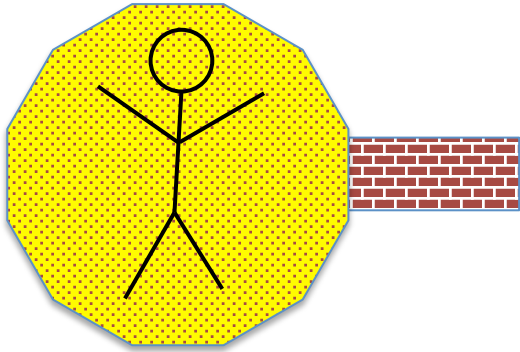


Speaker

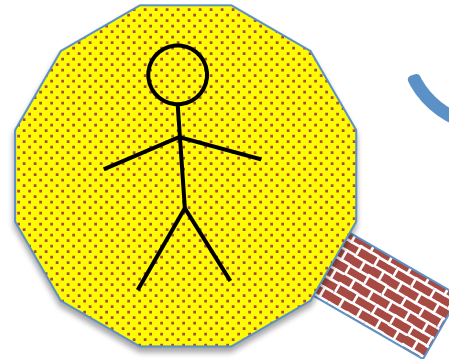
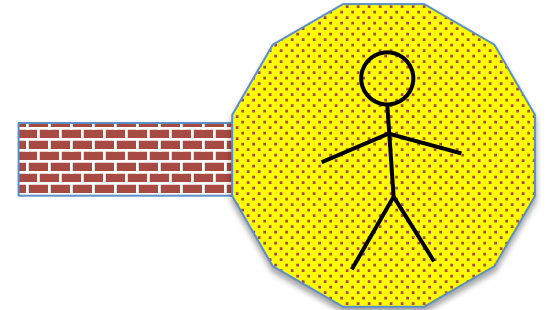
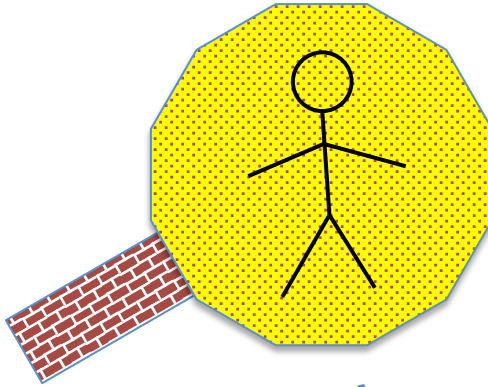


Audience

a bad talk:



Speaker

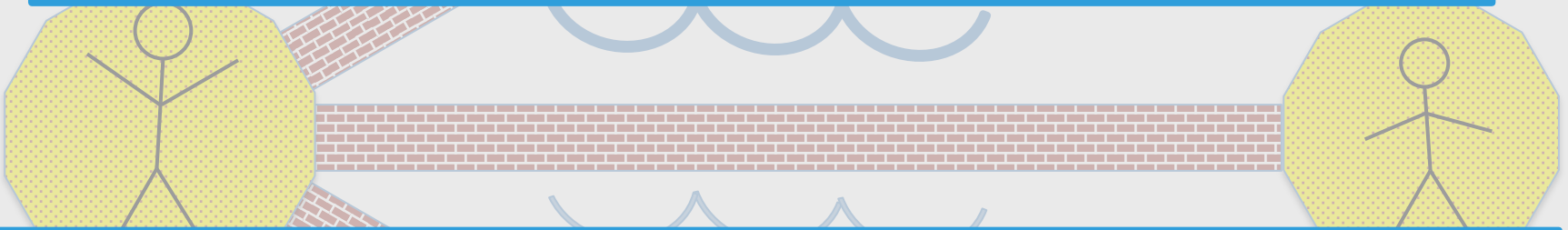


Audience



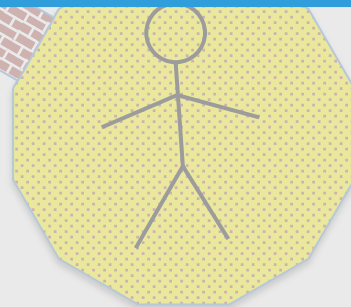
build bridges to your audience

organization, content, and delivery should be oriented to the audience



if you start building, they will help!

a good talk feels like a collaboration



Audience

Other Advice for Oral Presentations

watch videos by Jean-luc Doumont

- it's mostly common sense, yet we all forget!

to prepare an N -minute talk, you need $20N$ minutes

practice by yourself (out loud!) and with others

- this will help you to build bridges to everyone in the audience

for a conference talk, practice at least 10 times

What?

So what?



Information



Message

Interpretation

Get your audience to

- pay attention to,
- understand,
- (be able to) act upon

a maximum of messages, given constraints

Thanks!