Using distance-dependent Chinese restaurant process prior for unsupervised morphosyntactic clustering

Kairit Sirts

LTG Seminar 24.08.2015

The Scene

- Probabilistic modeling
  - Probability of the data given the model
- Clustering task
  - Probability of a data item belonging to a cluster
  - The number of clusters is unknown
- Bayesian model
  - Prior distribution over clustering models

Number of clusters

- Predefined number of clusters K
    - Discrete probability distribution over K items
    - For instance, for K = 3:
        - Uniform: [1/3, 1/3, 1/3]
        - Non-uniform: [0.7, 0.2, 0.1]
- Unknown number of clusters
    - We still want a discrete distribution over clusters
    - What should be the dimensionality of this distribution?

Outline

- Chinese restaurant process (CRP)
- Distance-dependent CRP (ddCRP)
- Morphosyntactic clustering with ddCRP

CRP metaphor

- Imagine …
  - … an infinitely big Chinese restaurant …
  - … with infinitely many tables …
  - … where each table is infinitely big accommodating infinitely many customers.
- At first the restaurant is empty.
- Then customers start coming one by one and …
  - … each customer sits into one of the already occupied tables with probability proportional to the number of customers already sitting there …
  - … or chooses to sit into an empty table with probability proportional to a predefined parameter.

Chinese restaurant process

- CRP is a stochastic process that generates discrete distributions

- Each infinite customer sequence defines a probability distribution over tables

- These distributions are infinite dimensional

- However, with N data points, only max N tables can be occupied

More formally

$$P(z_i = k | z_1, \ldots z_{i-1}; \alpha) = \begin{cases} \frac{n_k}{i-1+\alpha} & \text{if } k \in \{1, \ldots, K\} \\ \frac{\alpha}{i-1+\alpha} & \text{if } k = K+1 \end{cases}$$

k=1	k=2	k=3	k=4
n_1=2	n_2=1	n_3=1	n_4=0
$P(z_5 = 1) = \frac{2}{4+\alpha}$	$P(z_5 = 2) = \frac{1}{4+\alpha}$	$P(z_5 = 3) = \frac{1}{4+\alpha}$	$P(z_5 = 4) = \frac{\alpha}{4+\alpha}$

What do they eat?

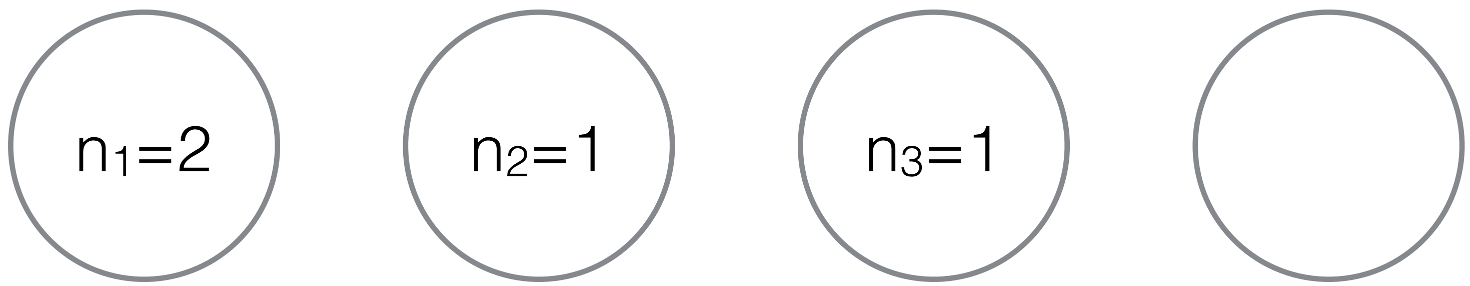
- The restaurant has a menu, which is related to a probability distribution P_0 (base distribution)
- The first customer in each table chooses a dish from the menu to be served on that table
- Thus, the probability of sitting into an empty table and eating a dish θ is:

$$P(z_i = K + 1, x_i = \theta) \propto \alpha P_0(\theta)$$

Who are these people?

- Hot-tempered and social southern people or introverted and individualistic nordic people?
- The concentration parameter ***alpha*** determines the shape of the generated distribution
    - Small ***alpha*** leads to bigger and fewer tables
    - Large ***alpha*** leads to more and smaller tables

Does it matter who comes first?



- There are several possible orderings:
  - for instance 1 1 2 3

$$P(z_1 = 1, z_2 = 1, z_3 = 2, z_4 = 3|\alpha) = \frac{\alpha}{0+\alpha}\frac{1}{1+\alpha}\frac{\alpha}{2+\alpha}\frac{\alpha}{3+\alpha}$$

- When we change the order of the customers:
  - The nominator stays the same
  - The terms in the numerator will be permuted
  - The overall joint probability will remain the same

Inference with Gibbs sampling

- Exchangeability - the joint probability does not depend on the order of the customers

- Exchangeability enables to use Gibbs sampling for inference.

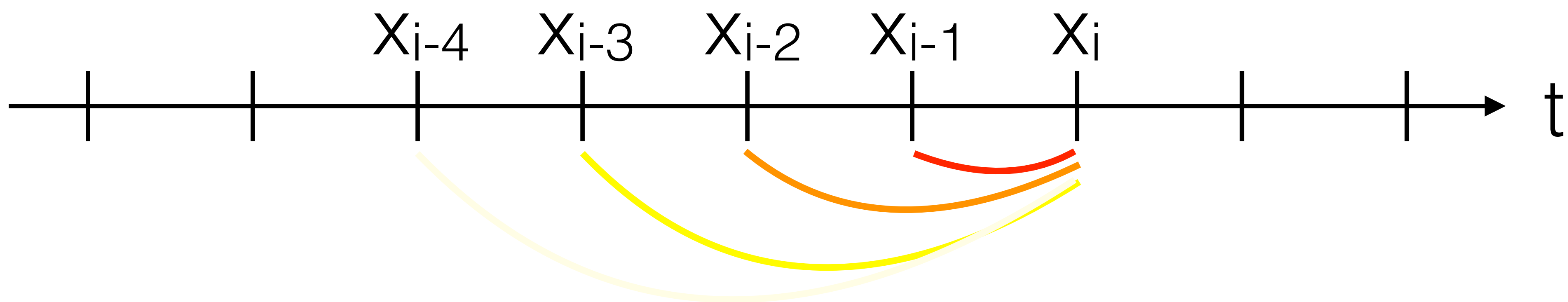
- Metaphorically works as follows:
    - Choose a customer and send him out
    - Clean the table if he was the last customer in that table
    - Pretend we've never seen this customer before
    - Ask him in as the next customer in the sequence and let him choose the table
    - Repeat with all customers many times

Distance-dependent Chinese restaurant process
Blei and Frazier, 2011

The story changes

- We still have an infinitely big restaurant …
- … and infinitely many tables with infinite capacity.
- Customers still come one by one.
- However, now each customer chooses to follow another customer …
- … proportional to the proximity or similarity to that customer.

The temporal setting



- The further away the data point the less likely we want to follow it

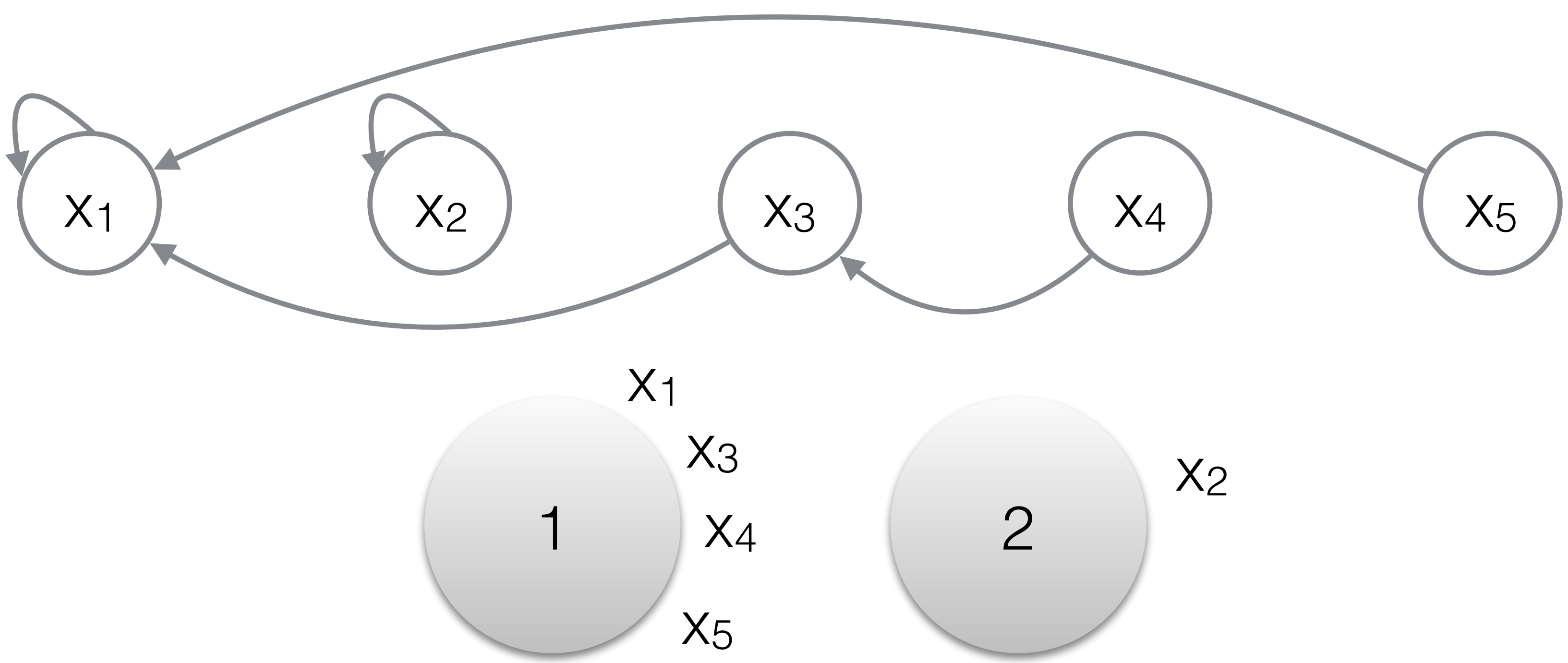
Formally

$$P(c_i = j | d, f, \alpha) \propto \begin{cases} f(d_{ij}) & \text{if } i \neq j \\ \alpha & \text{if } i = j \end{cases}$$

$$\begin{array}{ll} d & \text{distance matrix} \\ f & \text{decay function} \\ \alpha & \text{concentration parameter} \end{array}$$

- Alternatively, we may want customers to follow other customer they are most similar to.
- Combine distance and decay into a similarity function

Follower structure



- The follower structure defines the seating arrangement
- Several follower structures define the same seating arrangement
- *Sequential ddCRP* - all links point backwards

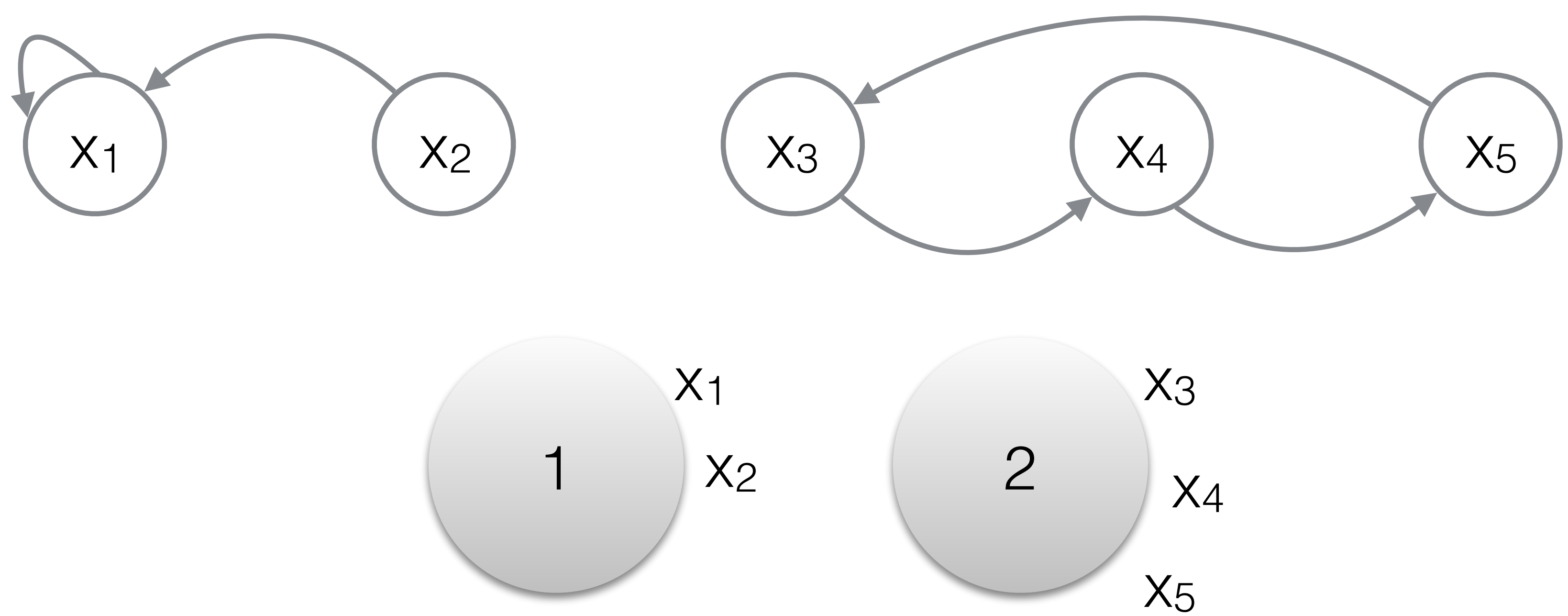
Relation to CRP

- Sequential ddCRP
- The similarity between all points is 1
- The probability of choosing any point to follow:

$$P(c_i = j|\alpha) \propto \begin{cases} 1 & \text{if } i \neq j \\ \alpha & \text{if } i = j \end{cases}$$

- The probability of sitting into particular table is the same as the CRP probability

Non-sequential ddCRP



- What if the links point forward?
- The generative story becomes messier
- Cycles can occur

Inference on the fly

- ddCRP is not exchangeable
- We still use Gibbs sampling
- We resample the ***links,*** not the table assignments
- Several scenarios can occur depending on the removed and added link properties
  - The seating arrangement will not change
  - The table will be broken into two
  - Two tables will be joined

Morphosyntactic clustering with ddCRP

Joint work with Jacob Eisenstein, Micha Elsner and Sharon Goldwater

The task

- Cluster together words with similar morphosyntactic function, e.g.
    - 3rd person present tense verbs: looks, walks, runs etc.
    - plural nominative case nouns: books, tables, floors etc.
    - present participle verbs: looking, walking, running etc

- Basically unsupervised POS clustering but in a more fine-grained level.

- Developed on English but the goal was eventually to apply it to morphologically more complex languages.

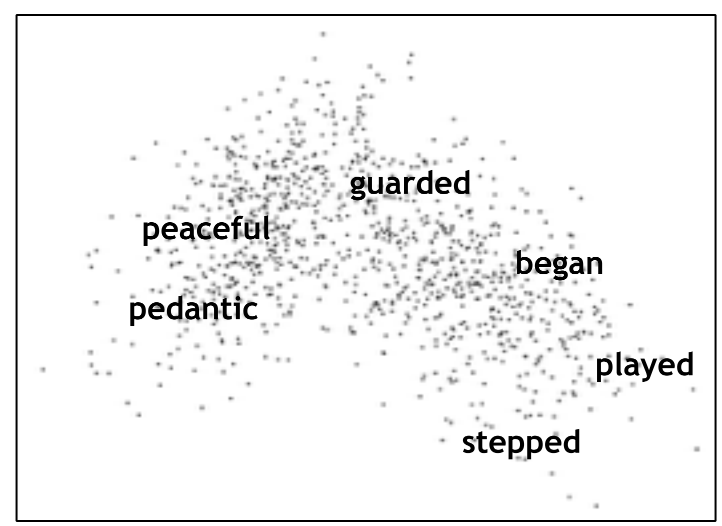
The model at a glance

- Uses two sources of information:
- Distributional information via word embeddings
  - Cluster the word embeddings with a Gaussian mixture model
- Morphological information via suffix features
  - Learn a suffix similarity function in the ddCRP prior

Word embeddings

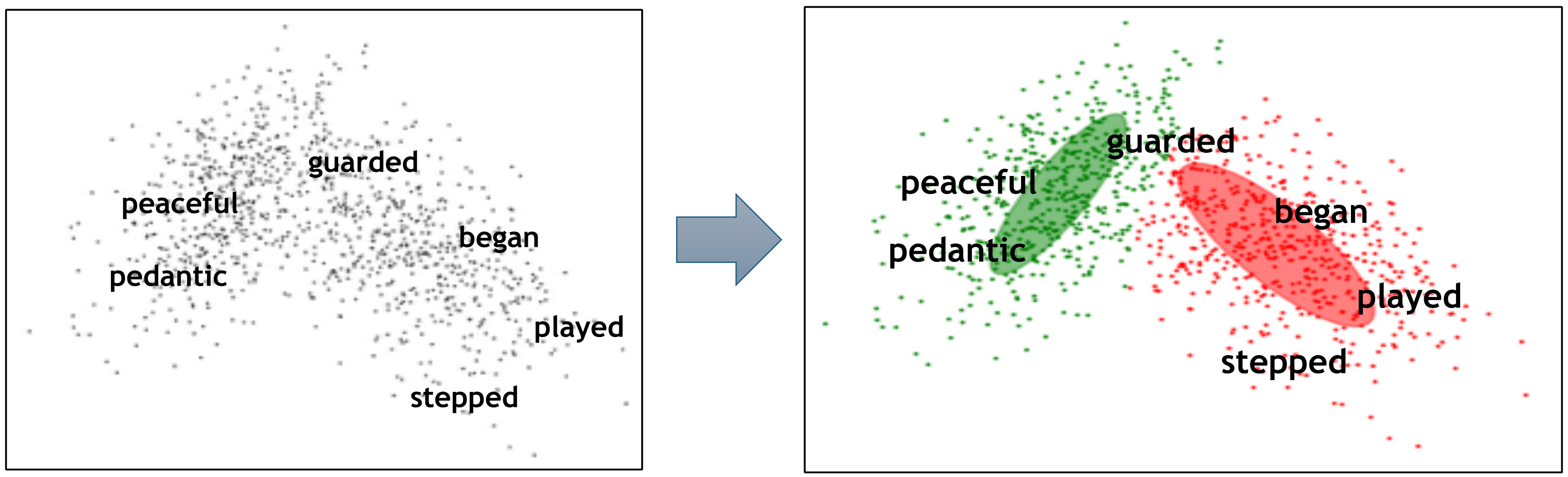
- Trained with neural network
- We used pre-trained Polyglot embeddings
  - Trained on Wikipedia
  - 100K most frequent words
  - 64-dimensional vectors

… and **began** copying …
… and **began** to …
…
… to **peaceful** sounds …
… on **peaceful** terms …
…

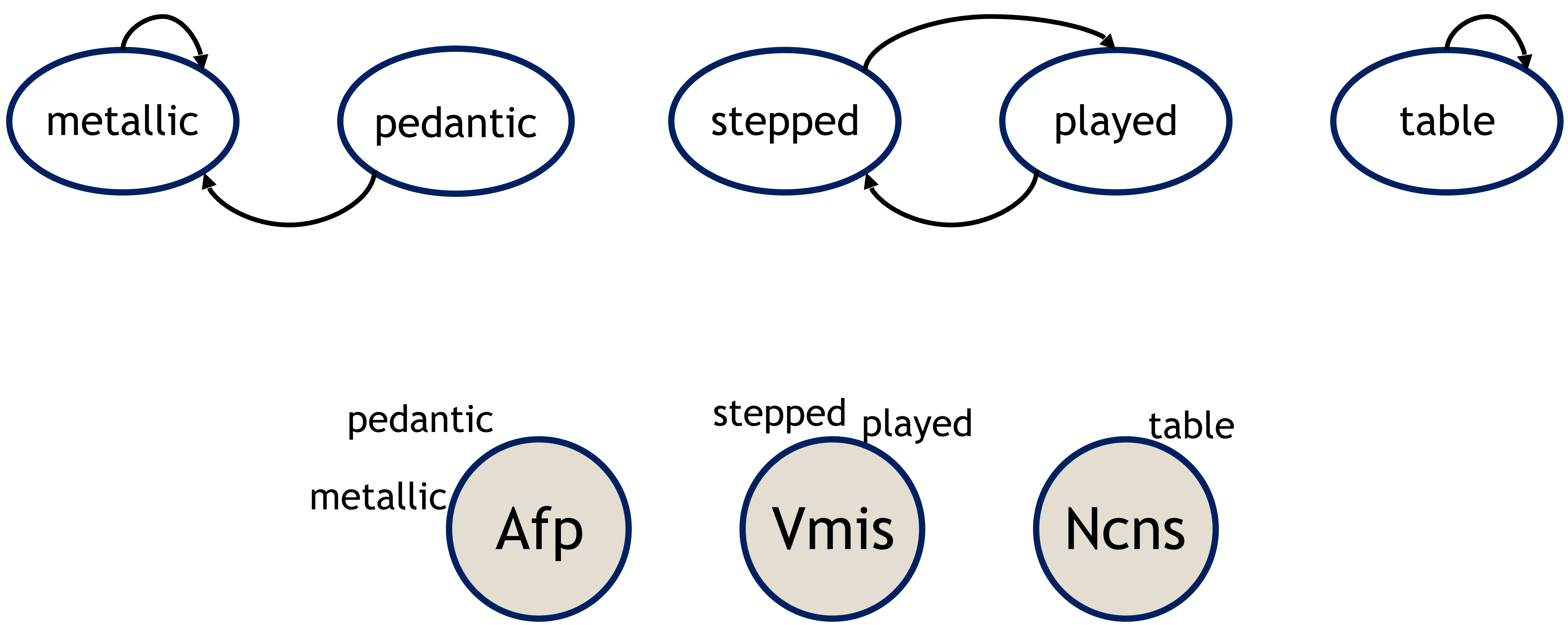


Gaussian likelihood

- Embeddings are treated as multivariate Gaussian random variables
- We fit a Gaussian mixture model with unknown means and co-variances
- The number of mixture components is not specified —> we need a non-parametric prior
  - CRP
  - ddCRP



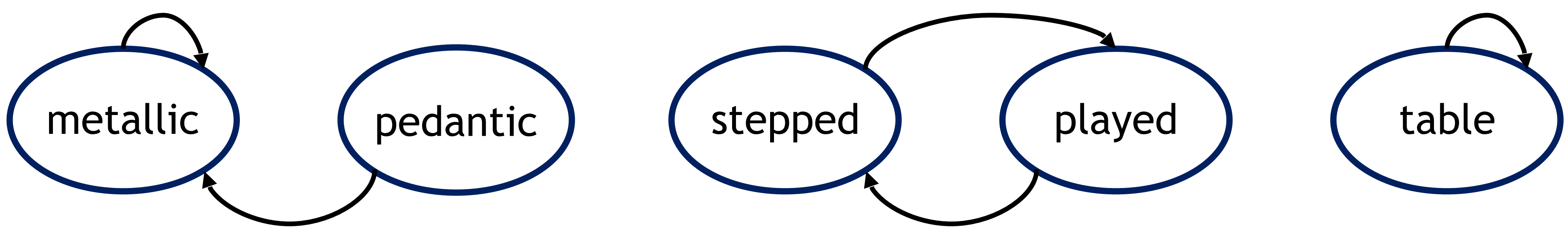
ddCRP prior



metallic
pedantic
stepped
played
table
pedantic
metallic
Afp
stepped
played
Vmis
table
Ncns

Similarity function

- Define the similarity between two words with a feature-based log-linear model



$$P(\text{stepped} \rightarrow \text{played}) \propto e^{w^T f(\text{stepped, played})}$$

$$P(\text{table} \rightarrow \text{table}) \propto \alpha$$

Morphological features

- Each word pair is assigned a feature vector
- Suffix features with max 3 characters

Word pair	-	-d	-ed	-c	-ic	-s	-es	...
stepped, played	1	1	1	0	0	0	0	...
metallic, pedantic	1	0	0	1	1	0	0	...

Learning the similarity function

$$P(\text{stepped} \ \rightarrow \text{played}) \propto e^{w^T f(\text{stepped, played})}$$
$$P(\text{table} \ \rightarrow \text{table}) \propto \alpha$$

- Where do we get those weights?
- Learn iteratively during model training
- The current follower structure acts as “supervised” data
- The weights can be trained with standard optimisation methods

Putting it all together

- infinite Gaussian mixture model with ddCRP prior fitted on word embeddings
- Trained with Gibbs sampling
- ddCRP prior uses a log-linear suffix similarity function over word pairs
- Similarity function is learned using standard optimisation methods
- Similarity function is updated after every Gibbs sweep over the data using the current follower structure as labelled data

Experiments

- We conducted experiments on Multext-East English part
  - Collection of morphologically annotated G. Orwell “1984” in 10 morphologically complex Eastern European languages + English
  - 104 fine-grained tags for English
  - almost half of them contain a single word only

- We tried with other languages too but got negative results due to:
  - low quality of the word embeddings
  - probably too simple similarity function

Baselines

- K-means:
    - Uses distributional information only (word embeddings)
    - Fixed number of clusters
    - Each cluster is equally likely a priori
- Infinite Gaussian mixture model:
    - Bayesian Gaussian mixture model with CRP prior
    - Uses distributional information only
    - Non-uniform prior over clusters
    - The number of clusters will be inferred from the data

Results

- Relatively good results on English
- Not too impressive results on other languages

Model	K	1-1	K-means
K-means	104	16.1	-
IGMM	55.6	41.0	23.1
ddCRP	47.2	**64.0**	25.0

ddCRP in a social experiment?

- The high level idea
  - Take a group of people who don’t know each other
  - Collect information from individual interviews to form a “likelihood”
  - Similarity function is just based on personal sympathy
  - Put people into a restaurant or some other nice place and start “resampling” based on the “likelihood” and the current subjective sympathy ranking
- How quickly would the configuration converge?
- Do the sympathies overlap with the best matches according to “likelihood”?

Thank you!
Questions?