# Week 7 report

**Donglai Wei**

2010.10.19

## 1) Reorganization of ME moves

### A) Focus on Dish Config

Inspired by Blei's "Reading Tea Leaves: How Humans Interpret Topic Models":
**Focus on the latent space:**
1) How well is a certain restaurant explained by the dishes ?$\Rightarrow$ Decompose-Restaurant Move(lower level)
2) How representative are the dishes? $\Rightarrow$ Dish Refinement(higher level)

So, we are going to do higher-level moves, directly refining dish config.

### B) Change

**Previously:**
a) We decompose a dish with the combo: Decompose-Res(Re-initialization)+Merge-Dish+Local-Dish
b) Accept/Reject at the end of the combo
**Now:**
a) Treat Re-initialization, Merge-Dish and Local-Dish equally as three moves to refine dish config
b) Accept/Reject at the end of each move

### C) Efficiency

Basically, the optimization problem has two phases:
1) Annealing:from **Bad** to **Good**
2) Search Until convergence: from **Good** to **Convergence**

(I) For annealing: we may want to keep the previous order, still doing the combo but Accept/Reject by themselves

(II) For Search until convergence:

  (1) In Decompose-Restaurant, we may only want to re-initialize the table with the certain dish. Otherwise, it may be too wasteful to throw away previous config which may be almost good.

  (2) In Decompose-Dish, we may want to decompose noiser dishes more frequently.

  (3) In Merge-Dish, we may want to merge smaller dishes more frequently.

## C) Pseudocode

$\%a) Annealing:$

```
For iter=1:n
    Temperature=(iter/n)^p
    For dish in dishes (random order):
            Decompose-Restaurant(restaurants serving dish)
            Local-Word(randsample(dish))
            Merge-Dish(randsample(dish,weight))
    End
End
```

$\%b) Run\ for\ Convergence:$

```
While Likelihood doesn't increase any more:
    Switch(randsample([1,2,3],weight_move))
        case 1:
        %1)Local Word Refinement () :
              Local-Word(randsample(dish))
        case 2:
        %1)Merge Dish Refinement () :
              weight~num of words in the dish
              Merge-Dish(randsample(dish,weight))
        case 3:
        %1)Decompose Dish Refinement () :
              weight~-Likelihood/num of words in the dish
  Decompose-Dish(randsample(dish,weight))
End
    End
```

## D) Tests

1) 200 restaurants, each of which is a 5 by 5 matrix(25 words) with 50 customers:
Initialization:
a) each restaurant forms its own dish
b) Simulation of the HDP-CRF: given $\alpha, \gamma$, we can first sample $t_{ji}$ for each restaurant and sample $k_{jt}$ for each table

ME-anneal+ME-search-until-convergence gives perfect result
*(Though I have to pick the annealing scheme after several tests manually.)*

2) 200 restaurants, each of which is a 10 by 10 matrix(100 words) with 200 customers:
Initialization:
a) Gibbs Block Sampler

ME-search-until-convergence gives perfect result

# 2) Gibbs Sampling with FAIR INITIALIZATION

Previously, I was scared away by the use of matlab class in Teh's code.

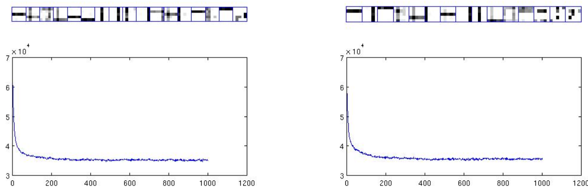

Figure 1: 5 by 5 matrix left: initialization a); right: initialization b)

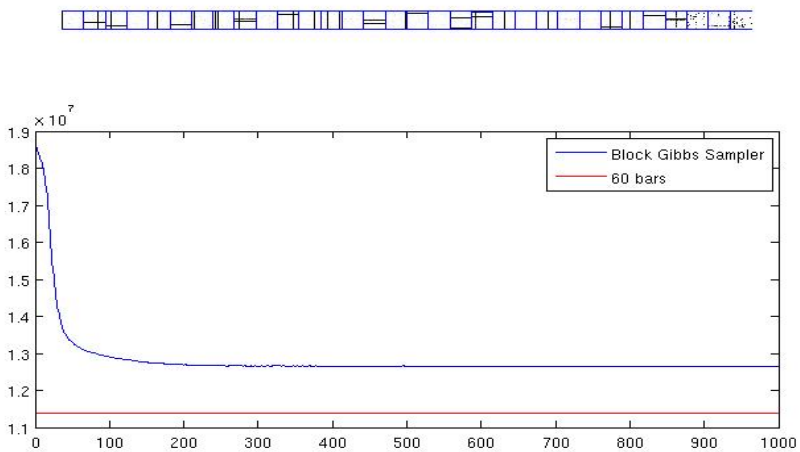


Figure 2: 30 by 30 matrix The Gibbs Block Sampler Breaks down

I worked out the test for npbayes-r1(1st version) while npbayes-r21(2nd version) is still a little too tricky for me figure out.

**Fair Initialization:**

(a) Simulation of the HDP-CRF: given $\alpha, \gamma$, we can first sample $t_{ji}$ for each restaurant and sample $k_{jt}$ for each table

(b) Naively, each restaurant forms its own dish.

1) I did the same test mentioned in D.1)for Gibbs Block Sampler(Teh's best sampler).
*(Namely, 200 restaurants, each of which is a 5 by 5 matrix(25 words) with 200 customers.)*
Generally, the results become much noiser than their own LDA-initialization. 2) I run it for a larger data set: 1000 restaurants, each of which is a 30 by 30 matrix(900 words) with 1800 customers.)
It only finds 30+ dishes, many of which are either noisy or composition of bars.

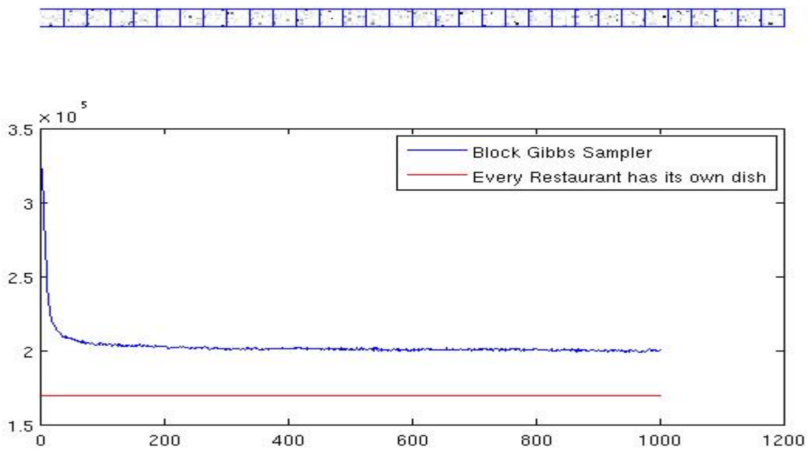


Figure 3: Gibbs Block Sampler on the synthetic topic data

# 3) Results on Topic Model

Here, I run Gibbs Sampling and ME algorithm on a synthetic data from NIPS abstract: 100 words and 200 documents, each of which has around 200 words.
*Similar size to that of previous bar test, but it is much more harder*

## a) Gibbs Sampler totally falls

Given the documents, intuitively, topic selection sucks if:

1. The algorithm gives too many topics
2. Each topic is a noisy combination of different words

Gibbs CRF sampler (though we know already it's no good) gives 178 topics.
Though Gibbs Block sampler gives only 32 topics, the average number of different types of words for each topic is 41.

## b) ME annealing

1) If we directly run ME after Gibbs Block Sampler, the likelihood will go up by making each restaurant a table, which is typical bad situation when the dish config is unclear and the gain only comes from maximizing t-term.

2) So we have to run annealing version of ME algorithm. When T is small, Decompose-Restaurant tends to have more restaurants. With the increase of the Temperature, t-term and k-term gradually come to the balance.

It's still running....taking so long...