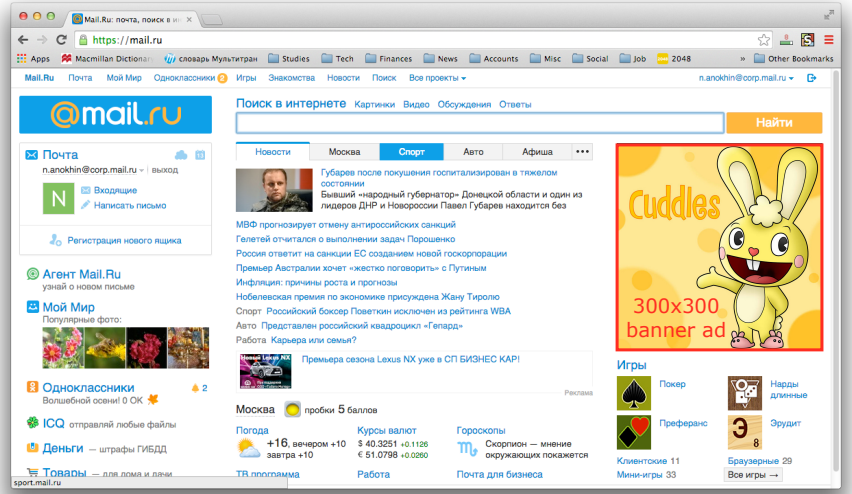
Using browsing history to identify Internet users' interests

Nikolay Anokhin
data scientist

@mail.ru group

2014 CEE-SEC(R) Software Engineering Conference in Russia

Advertisement on the Web



It's all about users (and money)

@mail.ru group

- ✗ clothing
- ✗ travelling
- ✗ cars
- ✗ dating

- ✓ computers
- ✓ gadgets
- ✓ photography
- ✓ data mining

The Data: user access logs

@mail.ru group

User ID	Timestamp	URL	Etc.
A1B2C3D4	2014-07-01 13:11:37	http://auto.mail.ru/toyota	M/27/...
A1B2C3D4	2014-07-01 13:20:45	http://example.com?id=football	M/27/...
A1B2C3D4	2014-07-02 00:25:10	http://somesite.com/index.php	M/27/...
...			
F9E8D7C6	2014-06-30 18:01:12	http://my-little-pony.com/	F/19/...
F9E8D7C6	2014-06-30 18:10:51	http://afisha.mail.ry/twilight	F/19/...

Text log files – about 300 G/day (and growing)

Some immediate conclusions

@mail.ru group

User ID	Timestamp	URL	Etc.
A1B2C3D4	2014-07-01 13:11:37	http://auto.mail.ru/toyota	M/27/...
A1B2C3D4	2014-07-01 13:20:45	http://example.com?id=football	M/27/...
A1B2C3D4	2014-07-02 00:25:10	http://somesite.com/index.php	M/27/...

A1B2C3D4: auto, toyota, football, somesite

Latent Dirichlet Allocation[1]

@mail.ru group

- Let there be M users, each user u is represented by a bag of N_u tokens
- Let the number of *topics* (user interests) be given and equal to K

Generative model

I For each topic draw a topic distribution $\beta_k \sim \text{Dir}(\eta_k),\ k \in 1, \dots K$

II For each user $u \in 1, \dots, M$:

1 Draw the user's topic distribution $\theta_u \sim \text{Dir}(\alpha)$

2 For each potential token $t \in 1, \dots, N_u$:

2.1 Choose the token's topic assignment $z_{u,t} \sim \text{MultI}(\theta_u)$

2.2 Choose the token $w_{u,t} \sim \text{Mult}(\beta_{z_{u,t}})$

[1] Latent Dirichlet Allocation // Blei et. al.

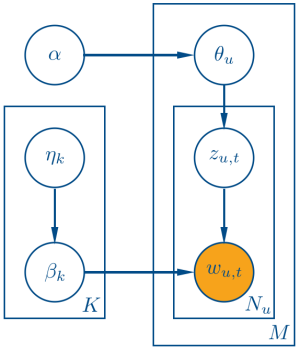
Generative model

$$p(\mathbf{w}, \theta, \beta, \mathbf{z}|\alpha, \eta) =$$

$$= p(\theta|\alpha) \prod_{t=1}^{N} p(z_t|\theta) p(w_t|z_t, \beta) p(\beta|\eta)$$

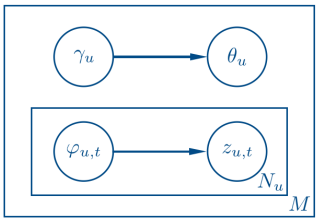
Posterior of hidden variables

$$p(\theta, \beta, \mathbf{z}|\mathbf{w}, \alpha, \eta) = \frac{p(\theta, \beta, \mathbf{z}, \mathbf{w}|\alpha, \eta)}{p(\mathbf{w}|\alpha, \eta)}$$



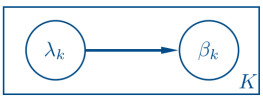
Variational inference

$$q(\theta, \beta, \mathbf{z}) = \prod_{k=1}^{K} \text{Dir}(\beta_k|\lambda_k) \times \\ \times \prod_{u=1}^{M} \text{Dir}(\theta_u|\gamma_u) \prod_{t=1}^{N} \text{Mult}(z_{u,t}|\varphi_{u,t})$$



Maximizing the ELBO...

$$\mathcal{L} = E_q \left[\log(p(\mathbf{w}, \theta, \beta, \mathbf{z})\right] - E_q \left[\log q(\theta, \beta, \mathbf{z})\right]$$



...is the same as minimising KL-divergence

$$KL(q||p) = E_q \left[\log \frac{q(\theta, \beta, \mathbf{z})}{p(\theta, \beta, \mathbf{z}|\mathbf{w})}\right]$$

Variational EM[2]

E1 For each user, given α and λ, update φ and γ

$$\varphi_{t,k} \propto E_q[\beta_{t,k}] \exp\left(\Psi(\gamma_l)\right)$$

$$\gamma_k = \alpha_k + \sum_{w=1}^{N} \varphi_{t,k}$$

E2 Update λ for each topic, using the obtained φ

$$\lambda_{t,k} = \eta_{t,k} + \sum_{u=1}^{M} w_t^{(u)} \varphi_{t,k}^{(u)}$$

M Maximise lower bound of the data log likelihood w.r.t. to α using Newton-Raphson method

[2]Mr. LDA: A Flexible Large Scale Topic Modeling Package using Variational Inference in MapReduce // Zhai et. al.

LDA – map

```
1  function configure()
2      load alpha, lambda and gamma from distributed cache
3      normalize lambda for every topic
4
5  function map(u, tokens)
6      initialize a zero V x K-dimensional matrix Phi
7      initialize a zero K-dimensional row vector sigma
8      read user logs as tokens w[1], w[2], . . , w[N]
9      repeat
10         for all t in 1..V do
11             for all k 1..K do
12                 Update Phi[t,k] = lambda[t,k]/(sum_t lambda[t,k]) * exp(Psi(gamma[u,k]))
13             normalize phi[t,*]
14             sigma = sigma + w[t] * phi[t,*]
15         Update row vector gamma[u,*] = alpha + sigma
16     until convergence
17     for all k in 1..K do
18         for all t in 1..V do
19             emit <k, t> : w[t] * phi[t,k]
20     emit <k, u> : gamma[u,k]
```

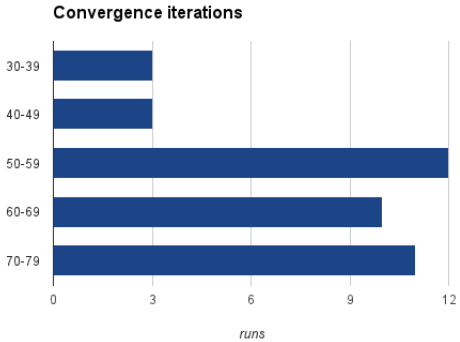
LDA – reduce

@mail.ru group

```
1 function map(<p_left, p_right>, Sigmas)
2         # sigma is unnormalized lambda
3         S = sum(Sigmas)
4         emit <p_left, p_right>: S
```

Running LDA

Typical machine config	
processors	2 x Intel(R) Xeon(R) 2.00GHz
cores	12
threads	24
RAM	32 GB
HDD	4-8 TB
30 machines in cluster	



Typical data: 10-days user logs
Typical run time: 20 hours

Modelling results – topics

@mail.ru group

topic1	topic2	topic3	topic4	topic5	topic6
book	klass	mobile	avito	krasnoyarsk	china
books	reshebnik	svyaznoy	kvartiry	tyumen	meta
loveread	class	phone	doma	tomsk	shared
knigi	megabotan	telefony	prodam	kemerovo	links
read	resh	nokia	dachi	surgut	maincat
author	slovo	phones	kottedzhi	barnaul	linkwall
litmir	algebra	iphone	nedvizhimost	nizhnevartovsk	nakanune
labirint	yazyk	samsung	sdam	krsk	razvezlo
authors	reshebniki	catalog	oblast	novokuznetsk	poster
tululu	otbet	allnokia	komnaty	kurgan	readme

Conclusions and Future Work

@mail.ru group

- LDA is an appropriate model for Internet user's interests
- Variational EM is an efficient algorithm for LDA parameter estimation
- Variational EM is easy to parallelise using MapReduce paradigm

- Profile prediction for a new user
- Topics as features in data mining tasks

Q&A

Nikolay Anokhin
n.anokhin@corp.mail.ru