COLLABORATIVE TOPIC MODELING FOR MOVIE RECOMMENDATION

Udbhav Prasad†, Abhishek Bhowmick†, Satwik Kottur††

udbhavp, abhowmi1, skottur @andrew.cmu.edu

Mentor: Emmanouil Antonios Platanios

†Computer Science Department, ††Electrical and Computer Engineering

Carnegie Mellon University

INTRODUCTION

Collaborative Filtering:

- Latent factor models work well in recommender systems
- Only useful in recommending "seen" items

Our Goal:

- Combine collaborative filtering with content analysis
- Latent themes from movies' plot summaries help to avoid the 'cold-start' problem in absence of item ratings

Motivation:

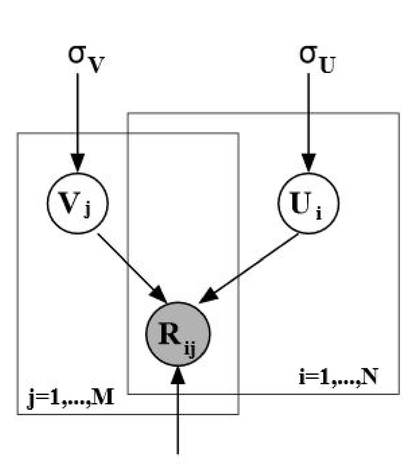
- Spotify, Pandora etc. use content analysis to recommend playlists
- Wang, Blei have tried this successfully with scientific articles [3]
- What about movie summaries ?

MATRIX FACTORIZATION

Idea:

- Exact inference is intractable
- Users / items represented by unobserved factors

Model:



(Figure) Graphical Model

R_{ij} rating of the i^{th} user for the j^{th} item.

$U \in R^{D \times N}$ - user matrix

$V \in R^{D \times M}$ - item matrix

$$p(R|U,V,\sigma^2) = \prod_{i=1}^{N}\prod_{j=1}^{M}\left[\mathcal{N}(R_{ij}|U_i^T V_j, \sigma^2)\right]^{I_{ij}}$$

$$p(U|\sigma_U^2) = \prod_{i=1}^{N}\mathcal{N}(U_i|0,\sigma_U^2\mathbf{I}), \qquad p(V|\sigma_V^2) = \prod_{j=1}^{M}\mathcal{N}(V_j|0,\sigma_V^2\mathbf{I})$$

Learning Algorithm:

- Maximizing the log-likelihood of the posterior over the item and user vectors is equivalent to minimizing

$$E = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}\left(R_{ij} - U_i^T V_j\right)^2 + \frac{\lambda_U}{2}\sum_{i=1}^{N}\| U_i \|_{Fro}^2 + \frac{\lambda_V}{2}\sum_{j=1}^{M}\| V_j \|_{Fro}^2$$

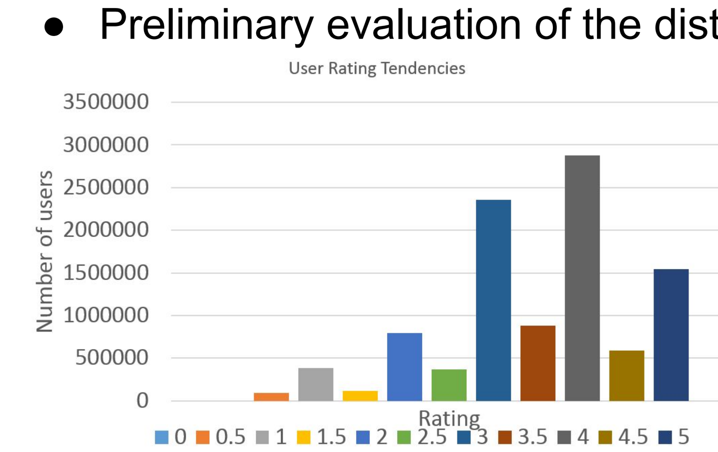
  which is the sum of squares objective error, with regularization terms.
- Also add biases for individual movies and users
- Apply gradient descent over movie and item vectors

Algorithm Design Decisions

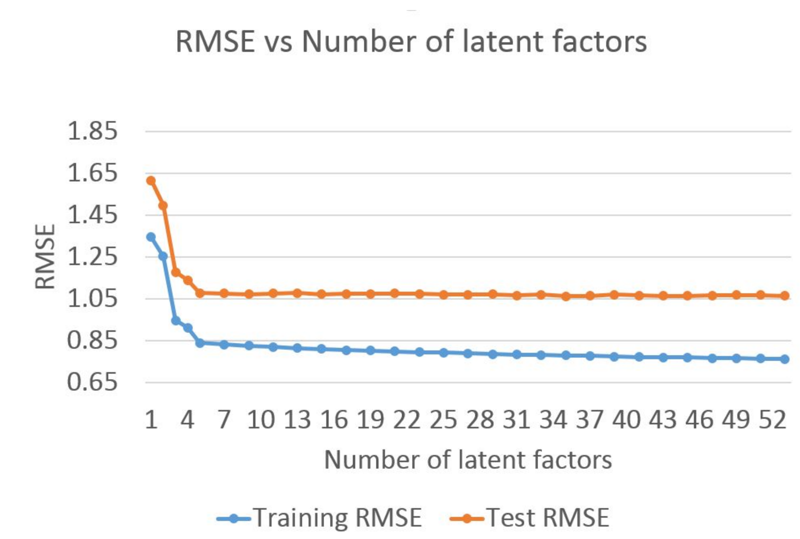
- Stochastic vs batch: Preferred stochastic descent owing to large amount of rating data.
- Used an adaptive learning rate.
- Learning rate parameters, regularization and number of latent factors were hyper-parameters of the model.
- Averaged gradient over previous iterations, and checked for convergence of the average.

Evaluation (PMF)

- Data set of 10 million movie ratings applied to 10,000 movies by 72,000 users (http://grouplens.org/datasets/movielens/)
- Preliminary evaluation of the distribution of ratings



Clearly the movie ratings are not centered around 2.5, so we needed to account for this bias.



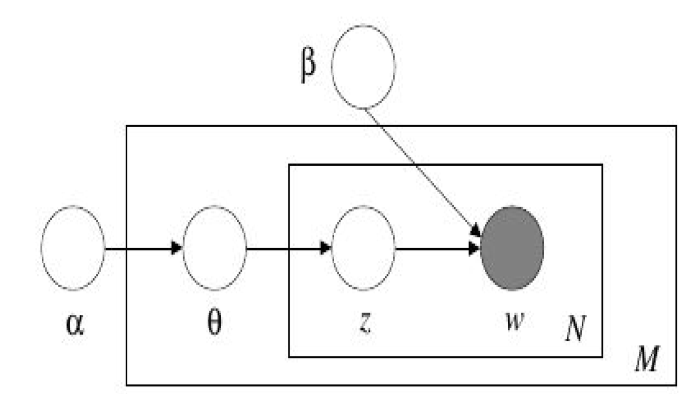
- Root mean squared error
- Hyper-parameter estimation
  - Grid search
  - Cross-validation
- λ = 0.01 and #factors > 10 seem to work well

LATENT DIRICHLET ALLOCATION

- Topic Modeling used to detect 'latent themes' in movie summaries
- Provides content-based representation of movies to be used for collaborative filtering

Generative Process

For each document D in corpus,
  Choose Θ ~ Dirichlet(α)
  For each word w_n in D
    Choose topic z_n~Multinomial(Θ)
    Choose w_n from $p(w_n| z_n, β)$



Graphical Model

- Variables are the topic mixture for each document - **Θ** {$Θ_1$, … , $Θ_D$}, topic distributions - **z** {z_1, … , z_n}. α, β are parameters of **Θ**, **z** respectively
- Joint distribution of **Θ, z** and **w** given α, β is :

$$p(\theta,\mathbf{z},\mathbf{w}\,|\,\alpha,\beta) = p(\theta\,|\,\alpha)\prod_{n=1}^{N}p(z_n\,|\,\theta)p(w_n\,|\,z_n,\beta)$$

(*exchangeability* - random variables z_n are independent conditioned on latent parameter Θ)

Inference

- To find posterior distribution of hidden variables given observed document $p(\theta,\mathbf{z}\,|\,\mathbf{w},\alpha,\beta) \rightarrow$ Use variational inference

Estimation

- Approximate empirical Bayes estimates through variational EM

Evaluation (LDA)

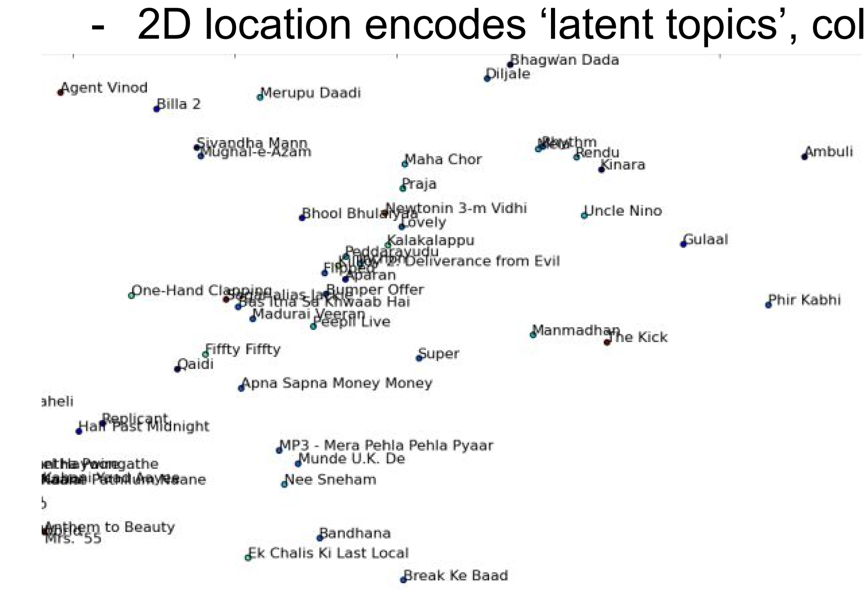
- Training data of 42,000 movie reviews from CMU Movie Summary Corpus (http://www.ark.cs.cmu.edu/personas/)
- Processed summaries used to learn 10-topic LDA using EM algorithm

Sample topics (with representative words)

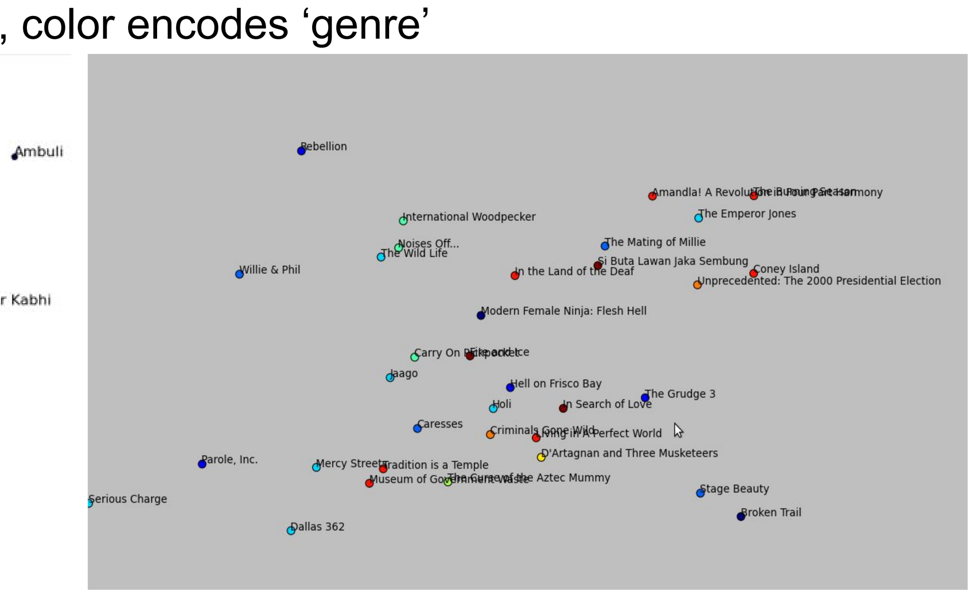
topic 1	topic 2	topic 3	topic 4	topic 5
school	war	police	earth	mother
game	army	gang	city	wife
friends	king	kill	crew	relationship
students	soldiers	prison	ship	children

Visualization of Movie Corpus

t-SNE [2] used to visualize feature representation of documents by LDA (in 3D)
- 2D location encodes 'latent topics', color encodes 'genre'



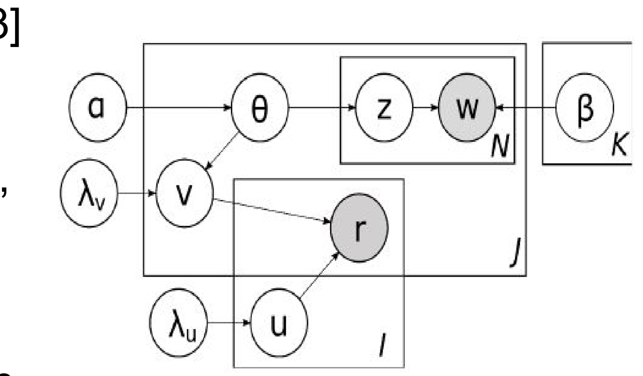
Indian Movies are clustered together



Documentaries based in America - Coney Island, Tradition is a Temple, Unprecedented : The 2000 Presidential Elections

COLLABORATIVE TOPIC REGRESSION

- Combine Collaborative Filtering & Topic Modeling [3]
- Replace item latent vector v_j as $v_j = e_j + Θ_j$
- Observed variables are **r** (ratings) & **w** (documents), latent variables are **Θ** (topic estimate), **z** (topic distribution), **v** (item vector) and **u** (user vector), parameters are α, β, $λ_u$ and $λ_v$
- Can perform both in-matrix and out-matrix prediction



FUTURE WORK

LDA:

- Implement and test inference procedure for unseen documents - Dirichlet smoothing for multinomial parameter β

Collaborative Filtering:

- Extend PMF to constrained PMF, to more accurately account for users with very few ratings
- Run the algorithm on the Netflix dataset, which has comparable standards, and is more representative of real user-item rating data

Collaborative Topic Regression:

- Implement CTR, evaluate in-matrix/out-of-matrix predictions

Stretch Goal

- Dynamic Topic Modeling to model user's preferences over time

REFERENCES

[1] David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. Latent dirichlet allocation. Journal of Machine Learning Research, 3:2003, 2003.

[2] L.J.P. van der Maaten and G.E. Hinton. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research 9(Nov):2579-2605, 2008.

[3] C. Wang, D. Blei. Collaborative topic modeling for recommending scientific articles. International Conference on Knowledge Discovery and Data Mining (KDD'11).

[4] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization.

[5] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems, 2009.