Using scores to improve language modelling of movie plot summaries

Jorge Sáez Gómez
Roelof van der Heijden
Francesco Stablum

Universiteit van Amsterdam

December 16, 2014

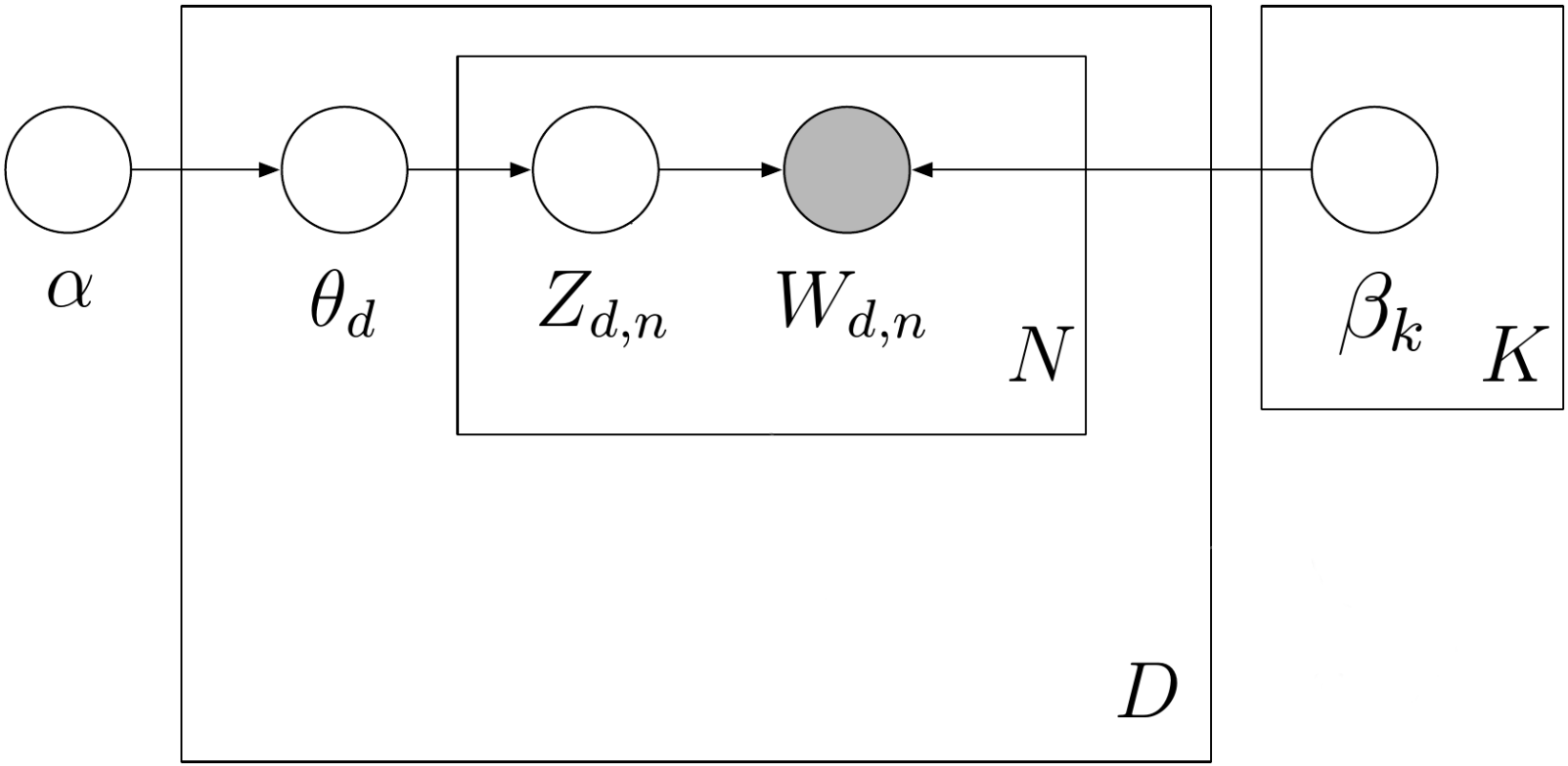
Presentation outline

Problem formulation

- Is there any correlation between the score of a movie and the contents of its script?
- Can we use the score to better model a movie corpus?

Models

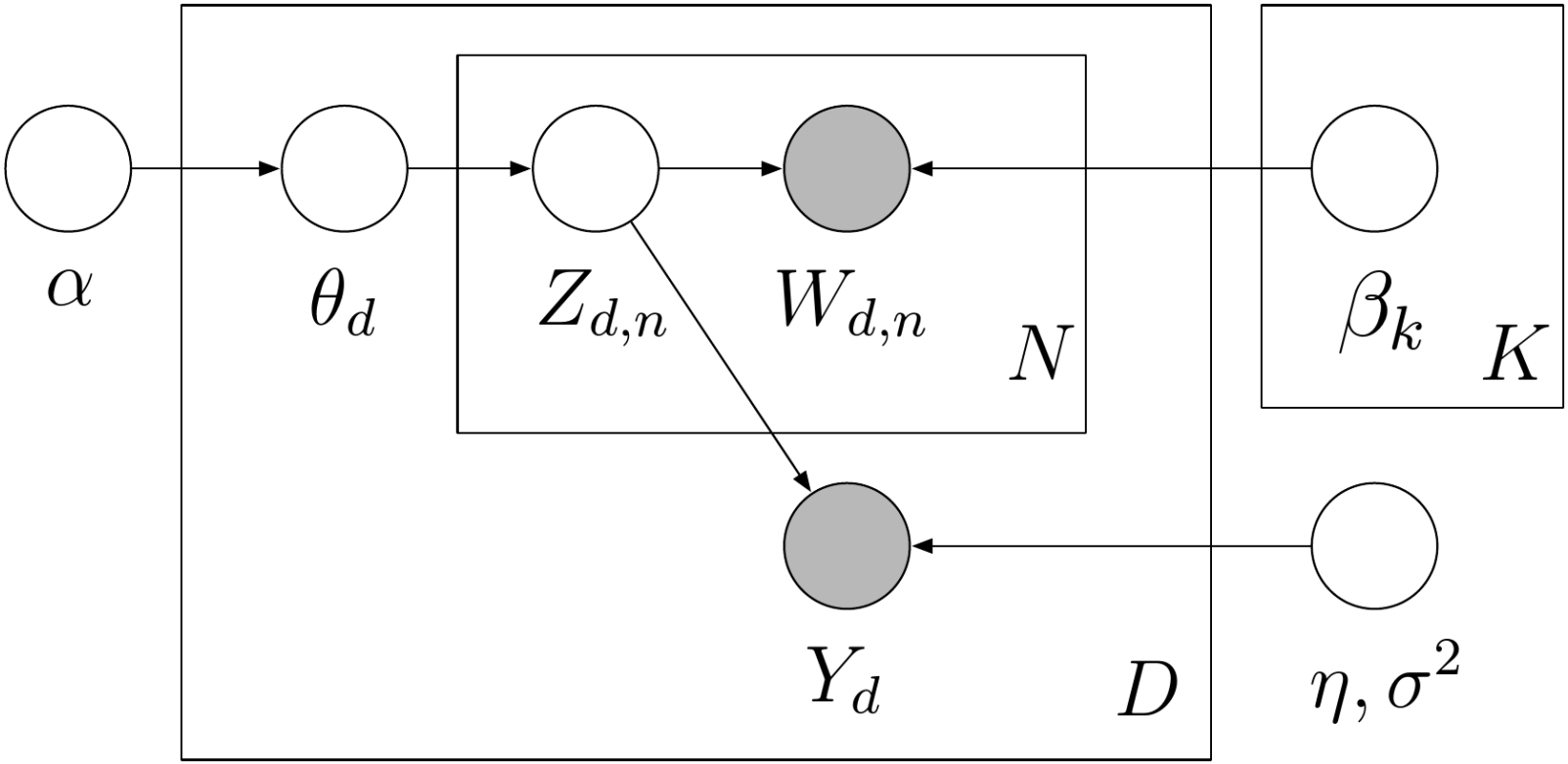
Latent Dirichlet Allocation:[1]



[1] Image taken from the paper "Supervised topic models" by David M. Blei and Jon D. McAuliffe (2007)

Models

Supervised Latent Dirichlet Allocation:[2]



[2] Image taken from the paper "Supervised topic models" by David M. Blei and Jon D. McAuliffe (2007)

Approach

Our collapsed Gibbs sampler:

$$p(z_{di} = k \mid Z^{\setminus i}, S, W, \alpha, \beta, \eta, \sigma) \propto \left[\prod_{k'} \frac{\prod_w \Gamma(N^{\setminus i}_{k'w} + \mathbb{I}(k' = k \wedge w = w_{di}) + \beta)}{\Gamma(N^{\setminus i}_{k'} + \mathbb{I}(k' = k) + W\beta)}\right] \times \underbrace{\mathcal{N}\left(s_d \,\middle|\, \eta^T \cdot \frac{N^{\setminus i}_{dk'} + \mathbb{I}(k' = k)}{N_d}, \sigma\right)}_{\text{Movie score term}} \prod_{k'} \Gamma(N^{\setminus i}_{dk'} + \mathbb{I}(k' = k) + \alpha)$$

Better implemented in log-space probabilities to avoid numerical problems.

Approach

Estimating the global score hyperparameter η:

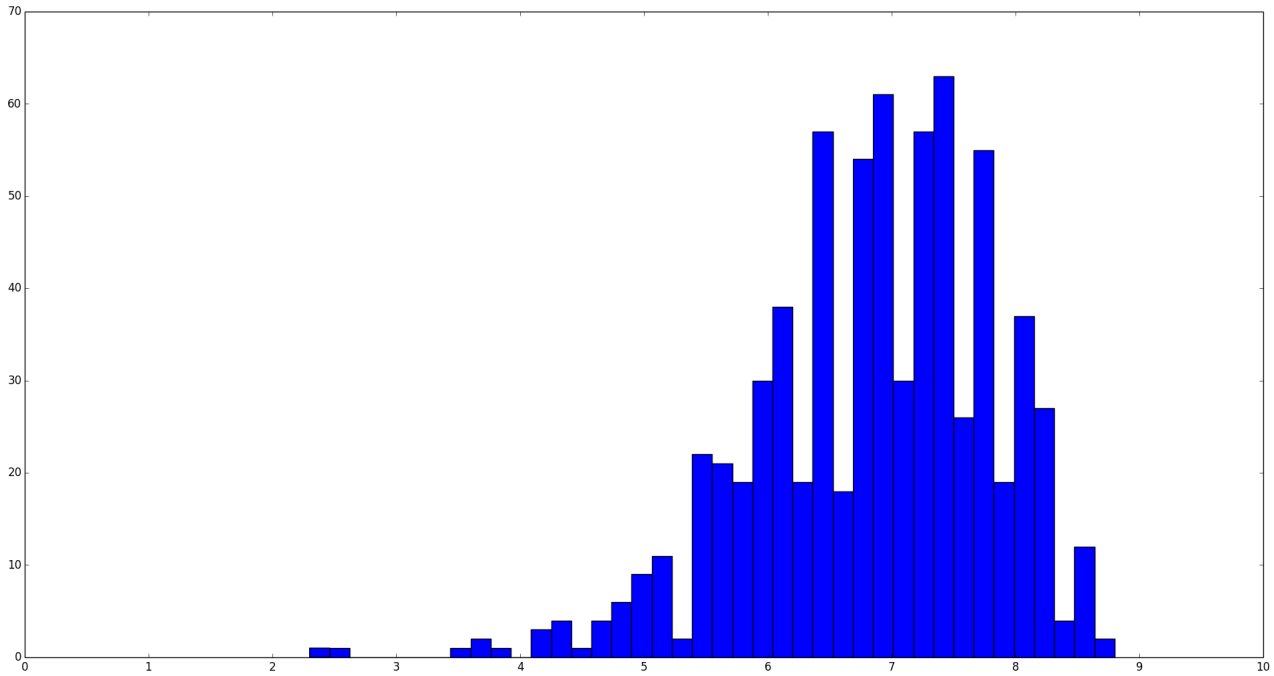
$$\eta_k^{new} \leftarrow (1-\gamma)\eta_k^{old} + \gamma \frac{\sum_d \frac{N_{dk}}{N_d}\left(s_d - \sum_{k' \neq k} \eta_{k'}^{old} \frac{N_{dk'}}{N_d}\right)}{\sum_d \left(\frac{N_{dk}}{N_d}\right)^2 + \varepsilon}$$

Where:

- $1 \gg \gamma > 0$ in order for the previous series to converge.
- $1 \gg \varepsilon > 0$ is a smoothing constant.

Dataset

- We made scripts to crawl `http://www.imsdb.com/` for movie scripts and then search `http://www.imdb.com/` for movie scores and plot summaries.
- We got a database with $\approx$ 700 movies.
- Movie score distribution (from 0 to 10):



Dataset

- Tokenization $\rightarrow$ stemming $\rightarrow$ pruning
- We prune words appearing only on a single movie (avoids overfitting) or within a stop list.
- Total number of tokens $\approx 12.7 \cdot 10^6$
- Number of unique tokens ≈ 35000
- Average number of tokens within a movie summary ≈ 75
- Average number of tokens within a movie script ≈ 18000

Results

- Initial selection of 120 movies (100 training / 20 testing) with balanced scores.
- Perplexity measure:

LDA topics		25	50	100
Using scores?	No	4128	5333	6954
	Yes	**4005**	5503	7082

- Inverse accuracy measure:

LDA topics		25	50	100
Using scores?	No	845	1318	2049
	Yes	**805**	1372	2067

Results

- Using 300 movies (250 training / 50 testing).
- Results averaged across 3 MCMC chains.
- Perplexity measure:

LDA topics		25
Using scores?	No	3237
	Yes	**2955**

- Inverse accuracy measure:

LDA topics		25
Using scores?	No	597
	Yes	**571**

Discussion

- Use of few movies $\Rightarrow$ the model starts overfitting with 50-100 topics already.
- Slight predictive improvement if using movie scores, but it is not significant.

Challenges

- Improve speed of the collapsed Gibbs sampler.
- Use the movie scripts instead of the movie summaries.
- Use both the movie scripts and summaries.
- Incorporate more information into the model, such as the movie genre.