Latent Dirichlet allocation in web spam filtering

István Bíró[1] Jácint Szabó[1] András A. Benczúr[1]

[1]Data Mining and Web Search Group
Computer and Automation Institute
Hungarian Academy of Sciences

AIRWeb Workshop, April 22, 2008, Beijing, China.

Latent Dirichlet allocation

- Blei, Ng, Jordan, 2003
- fully generative statistical natural language model
- extension of latent semantic indexing (LSI)
- has better perplexity than LSI
- a lot of extensions and variations of LDA were developed and successfully applied

Latent Dirichlet allocation

Model

- topic: distribution over the words
- document: distribution over the topics
- for every word-position of the corpus, draw a topic for that document, and then draw a word for that topic

Inference by Gibbs sampling

- Markov chain Monte Carlo method
- running time $O((\#\text{topics}) \cdot (\#\text{word pos's}) \cdot (\#\text{iterations}))$

Latent Dirichlet allocation

Model

- topic: distribution over the words
- document: distribution over the topics
- for every word-position of the corpus, draw a topic for that document, and then draw a word for that topic

Inference by Gibbs sampling

- Markov chain Monte Carlo method
- running time $O((\#\text{topics}) \cdot (\#\text{word pos's}) \cdot (\#\text{iterations})\ )$

Latent Dirichlet allocation

In practice

- given a collection of documents
- keep only semantic words, delete stopwords, stem
- create vocabulary
- choose an appropriate topic-number (about 100)
- make model inference to create the model
- for a topic, the word distribution gives a semantic theme
- for a document, the topic distribution describes to which themes it belongs
- for a new document, make unseen inference to get its topic distribution

Latent Dirichlet allocation

In practice

- given a collection of documents
- keep only semantic words, delete stopwords, stem
- create vocabulary
- choose an appropriate topic-number (about 100)
- make model inference to create the model
- for a topic, the word distribution gives a semantic theme
- for a document, the topic distribution describes to which themes it belongs
- for a new document, make unseen inference to get its topic distribution

Latent Dirichlet allocation

In practice

- given a collection of documents
- keep only semantic words, delete stopwords, stem
- create vocabulary
- choose an appropriate topic-number (about 100)
- make model inference to create the model
- for a topic, the word distribution gives a semantic theme
- for a document, the topic distribution describes to which themes it belongs
- for a new document, make unseen inference to get its topic distribution

Multi-corpus LDA

- two corpora: labeled spam and nonspam sites
- build two separate LDA models on them, with k_s and k_n topics
- aggregate these models to get $k_s + k_n$ topics
- make unseen inference for every unlabeled site, and get its topic-distribution: $\sum_{1 \le i \le k_s} p_i^s + \sum_{1 \le i \le k_n} p_i^n = 1$
- the total probability of spam topics, $\sum_{1 \le i \le k_s} p_i^s$, gives a spamness feature

Similar to the compression based spam filter of Cormack, applied with success at Web Spam Challenge 2007.

Tests

Multi-corpus LDA on UK2007-WEBSPAM, apparently primarily content spammed

- $\sim$ 115000 sites
- $\sim$ 200 labeled as spam
- $\sim$ 3800 labeled as nonspam

- document: concatenation of all pages of a site
- topic numbers: $k_s = 10$ and $k_n = 50$

Tests

Most frequent words in some topics

Spam topic 7

- loan (0.080)
- uk (0.042)
- unsecured (0.026)
- credit (0.024)
- home (0.022)

Nonspam topic 4

- club (0.035)
- team (0.012)
- league (0.009)
- win (0.009)
- home (0.009)

Nonspam topic 10

- music (0.022)
- band (0.012)
- film (0.011)
- festival (0.009)
- dance (0.008)

Tests

- public: Web Spam Challenge 2008 public features
- text: pivoted tf.idf (Singhal et al.)
- graph: site and page level stacked graphical (see Dávid Siklósi's Challenge talk)

feature set	F	ROC
text (SVM)	0.554	0.864
public & text & graph (log)	0.601	0.954
public & text & graph & lda (log)	0.667	0.969