MLG seminar

Fully Sparse Topic Models

Khoat Than & Tu Bao Ho

June, 2012

Content

- Introduction
- Motivation
- Research objectives
- Fully sparse topic models
- A distributed architecture
- Conclusion and open problems

Motivation

Large-scale modeling

- Recent practical applications:
  - Millions/billions of instances [Bekkerman et al. 2012].
  - Millions of dimensions [Yu et al. 2012, Weinberger et al. 2009].
  - Large number of new instances to be processed in a limited time [Bekkerman et al. 2012].
- Topic modeling literature:
  - Corpora of millions of documents [Hoffman et al. 2010]
  - Billions of terms in N-gram corpora or in IR systems [Wang et al. 2011]
  - Thousands of latent topics need to be learned [Smola et al. 2010]
- Learn a topic model which consists of billions of hidden variables.

Limitations of existing topic models

- Dense models:
  - Most models, e.g. LDA and PLSA, assume all topics have non-zero contributions to a specific document → unrealistic
  - This results in dense representations of data → huge memory for storage.
  - Topics are often assumed to follow Dirichlet distributions →dense topics
- High complexity:
  - Inference in LDA is NP-hard [Sontag & Roy, 2011]
  - Learning of topics is time-consuming.

Limitations: large-scale learning

- Learning LDA: online [Hoffman et al. 2010], parallel [Newman et al. 2009], distributed [Asunction et al. 2011]
  - Dense topics,
  - Dense latent representations of documents,
  - Slow inference.
- Regularized latent semantic indexing (RLSI): [Wang et al. 2011]
  - Learning and inference are triple in #topics →high complexity
  - Auxiliary parameters requires model selection → problematic
  - Cannot trade off sparsity against time and quality.

Research objectives

Objectives

- Develop a topic model:
  - Deal with huge data of million/billion dimensions.
  - Handily learn thousands of latent topics.
  - The learned topics and new representations of documents are very sparse.
  - Inference is fast, e.g., in linear time.

Fully sparse topic models

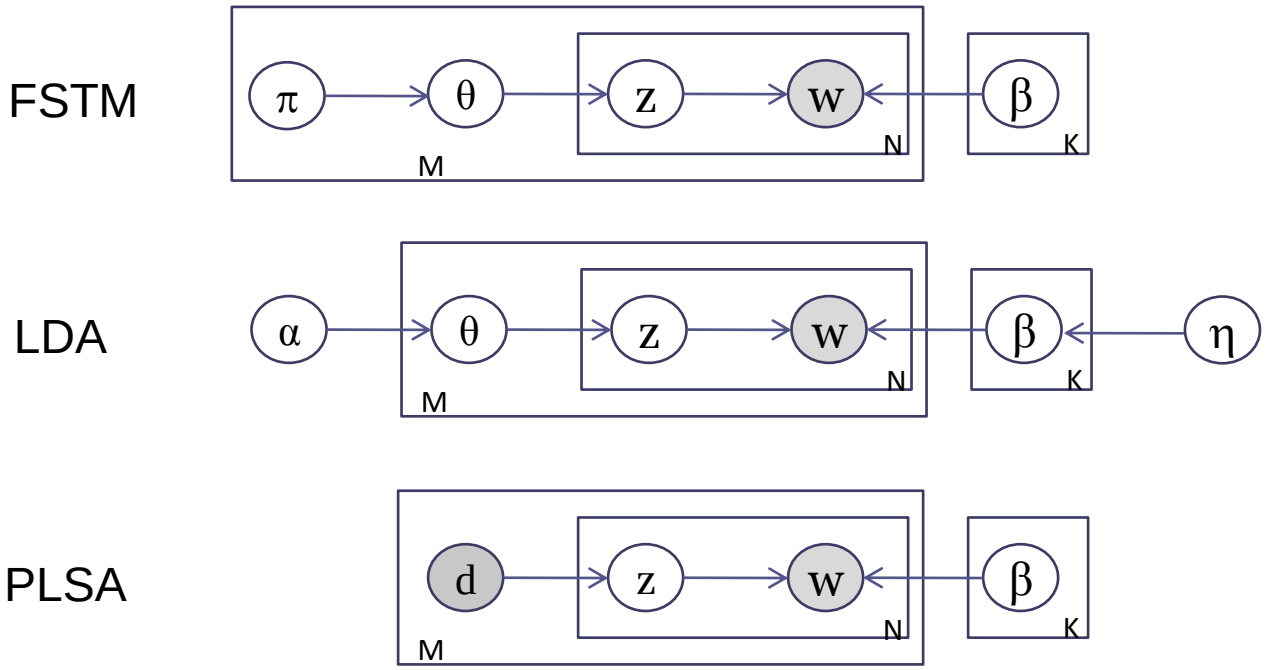
FSTM: model description

- We propose *Fully sparse topic models (FSTM)*.
  - FSTM assumes topic proportions in documents to be sparse.
  - It assumes a corpus to be composed of K topics,
  - Each document is a mixture of some of K topics.
  - Generative process of each document $\boldsymbol{d}$:

1. *Pick a subset* $\boldsymbol{\pi}$ *from* $\{1, 2, ..., K\}$.
2. *For the* j*th word in* $\boldsymbol{d}$*:*
   - *Pick a latent topic* $z_k \in \boldsymbol{\pi}$ *with probability* $P(z_k|\boldsymbol{d}) = \theta_k$,
   - *Generate a word* w_j *with probability* $P(w_j|z_k) = \beta_{kj}$.

FSTM: model description

- Graphical representation:



FSTM: scheme for learning and inference

- How to scale up the learning and inference?
  - Reformulate inference as a concave maximization problem over simplex of topics.
  - Sparse approximation can be exploited seamlessly.
  - Learning is formulated as an *easy* optimization problem.
  - Consequence: learning of topics amounts to multiplication of two very sparse matrices.

FSTM: inference

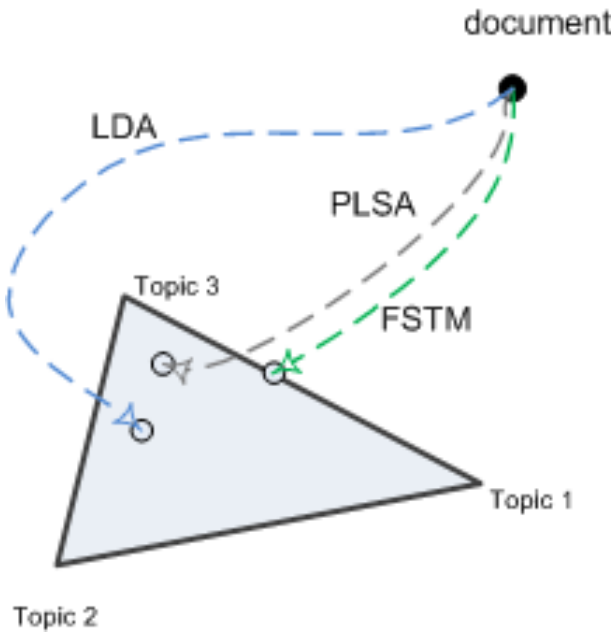
- Problem: given the model with K topics, and a document $\boldsymbol{d}$, we are asked to infer π and θ.
  - Reformulate inference as a concave maximization problem over simplex of topics.

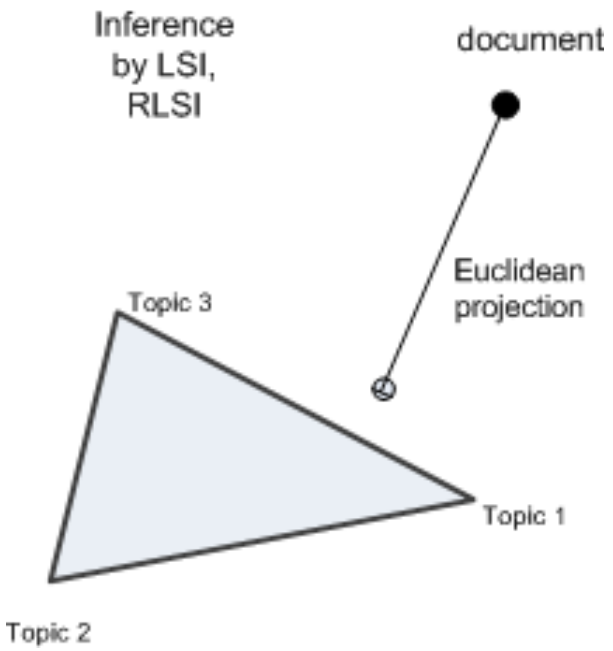
Lemma 1. *Consider FSTM with topics* $\beta_1, ..., \beta_K$, *and a given document* $\boldsymbol{d}$. *The inference problem can be reformulated as the following concave maximization problem, over the simplex* $\Delta = conv(\beta_1, ..., \beta_K)$,

$$x^* = \arg\max_{x \in \Delta} \sum_{j \in I_d} d_j \log x_j. \quad (3)$$

FSTM: inference

- Geometric interpretation
  - Inference is a projection onto the simplex of topics.
  - Different objective functions for projection are used.
  - FSTM always projects documents onto the boundary.





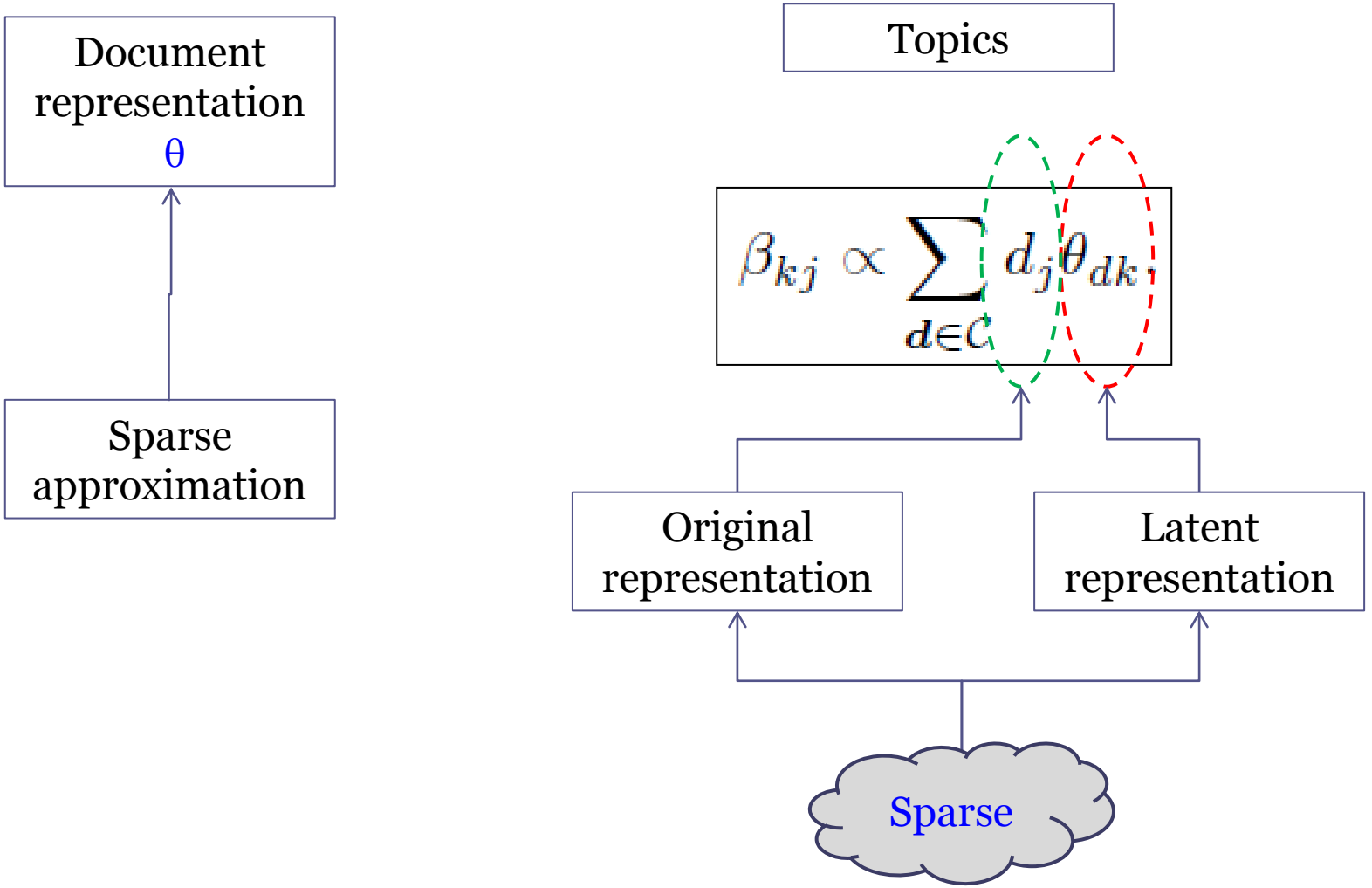
FSTM: learning and inference

- Learning from a corpus C: EM scheme
  - E-step: each document is inferred separately.
  - M-step: maximize the likelihood over topics $\beta_{kj} \propto \sum_{d \in C} d_j \theta_{dk}$.

Algorithm 1 Inference algorithm

Input: document d and topics $\beta_1, ..., \beta_K$.
Output: θ_*, for which $\sum_{k=1}^{K} \theta_{*,k}\beta_k = x_*$ maximizes $f(x) = \sum_{j \in I_d} d_j \log x_j$.
Pick as β_r the vertex of $\Delta = conv(\beta_1, ..., \beta_K)$ with largest f value.
Set $x_0 := \beta_r; \theta_{0,r} = 1; \theta_{0,k} = 0, \forall k \neq r;$
for $\ell = 0, ..., \infty$ **do**
  $i' := \arg\max_i \beta_i^t \nabla f(x_\ell);$
  $\alpha' := \arg\max_{\alpha \in [0,1]} f(\alpha\beta_{i'} + (1-\alpha)x_\ell);$
  $x_{\ell+1} := \alpha'\beta_{i'} + (1-\alpha')x_\ell;$
  $\theta_{\ell+1} := (1-\alpha')\theta_\ell;$ and then set $\theta_{\ell+1,i'} := \theta_{\ell+1,i'} + \alpha'.$
end for

FSTM: why sparse?



FSTM: theoretical analysis

- Theoretical analysis:
  - Goodness of inference

Theorem 2. *Consider FSTM with K topics, and a document $\boldsymbol{d}$. Let C_f be defined as in Theorem 1 for the function $f(\boldsymbol{x}) = \sum_{j \in I_d} d_j \log x_j$. Then Algorithm 1 converges to the optimal solution with a linear rate. In addition, after L iterations, the inference error is at most $4C_f/(L+3)$, and the topic proportion $\boldsymbol{\theta}$ has at most $L+1$ non-zero components.*

  - Inference quality can be arbitrarily approximated.
  - Sparsity is obtained by limiting the number of iterations.

FSTM: theoretical comparison

Existing topic models

- Inference is very slow.
- No guarantee on inference time.
- No bound for posterior approximation.
- Quality of inference is often not known.
- Impose sparsity via regularization techniques or incorporating some distributions → indirectly control sparsity.
- Sparsity level cannot be predicted.
- No principled way to trade off goodness-of-fit against sparsity level.

FSTM

- Inference is in linear time.
- Explicit bound for posterior approximation.
- Quality of inference is explicitly estimated and controlled.
- Provide a principled way to directly control sparsity level.
- Sparsity level is predictable.
- Easily trade off goodness-of-fit against sparsity level.

FSTM: theoretical comparison

Table 1. Theoretical comparison of some topic models. V is the vocabulary size, K is the number of topics, n is the length of the document to be inferred. $\bar{K}$ is the average number of topics to which a term has nonzero contributions, $\bar{K} \leq K$. L is the number of iterations for inference. $\bar{K}$ (and L) is different for these models. '-' denotes *'no'* or *'unspecified'*; '✓' means *'yes'* or *'taken in consideration'*.

Model	FSTM	PLSA	LDA	STC	SRS	RLSI
Document sparsity	✓	-	-	✓	✓	-
Topic sparsity	✓	-	-	-	✓	✓
Sparsity control	direct	-	-	indirect	indirect	indirect
Trade-off:						
sparsity vs. quality	✓	-	-	-	-	-
sparsity vs. time	✓	-	-	-	-	-
Inference complexity	$L.O(n.\bar{K} + K)$	$L.O(n.K)$	$L.O(n.K)$	$O(n.K)$	$L.O(n.K)$	$L.O(V.\bar{K}^2 + K^3)$
Inference error	$O(1/L)$	-	-	0	-	0
Storage for topics	$V.\bar{K}$	$V.K$	$V.K$	$V.K$	$V.\bar{K}$	$V.\bar{K}$
Auxiliary parameters	0	0	0	3	2	2

FSTM: experiments

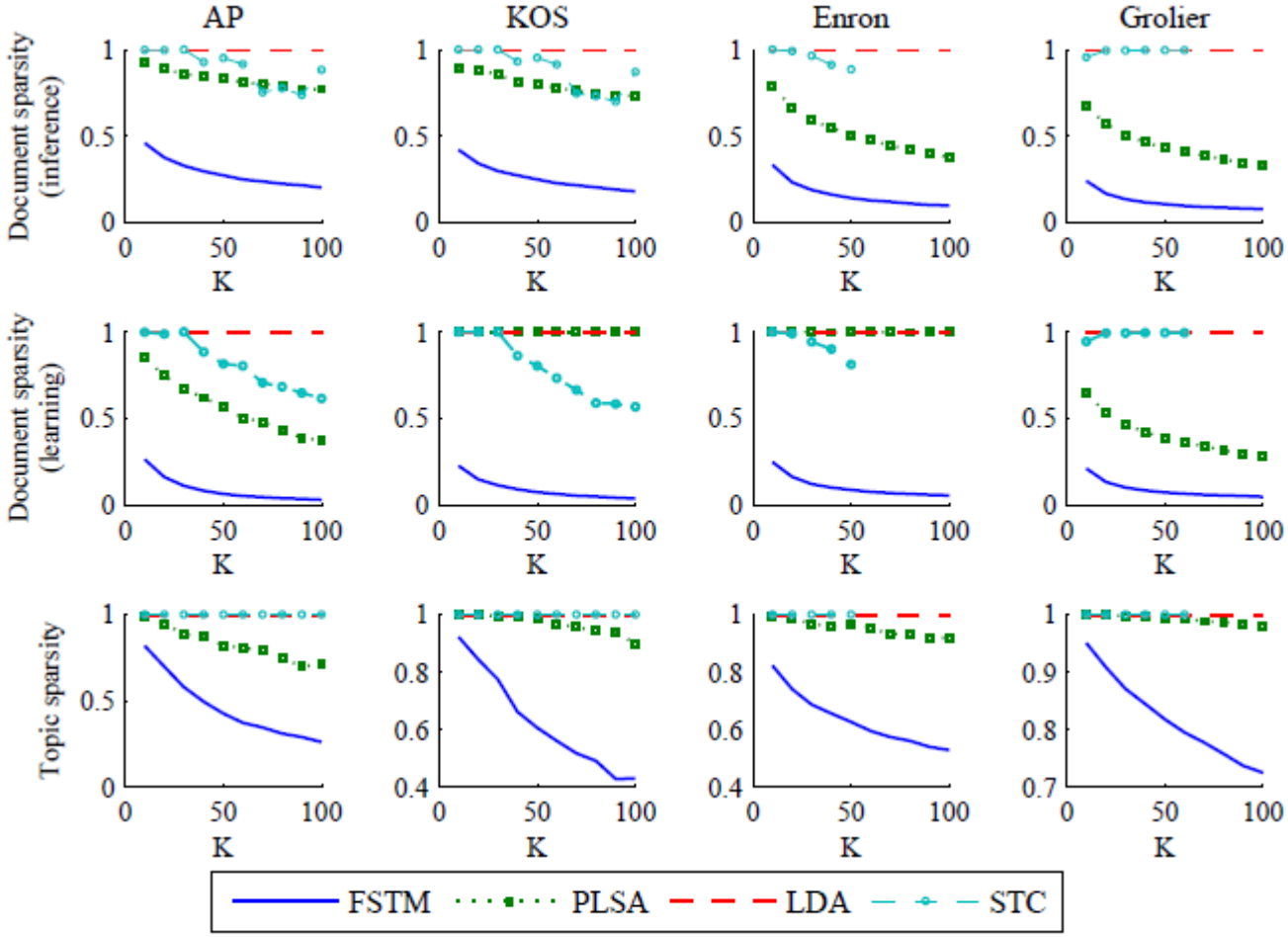
- Data:

Data	#Training	#Testing	#Dimension
AP	2,021	225	10,473
KOS	3,087	343	6,907
Grolier	23,044	6,718	15,276
Enron	35,875	3,986	28,102
Webspam	350,000	350,000	16,609,143

- Models for comparison:
  - Probabilistic latent semantic analysis (PLSA) [Hofmann, 2001]
  - Latent Dirichlet allocation (LDA) [Blei et al., 2003]
  - Sparse topical coding (STC) [Zhu & Xing, 2011]

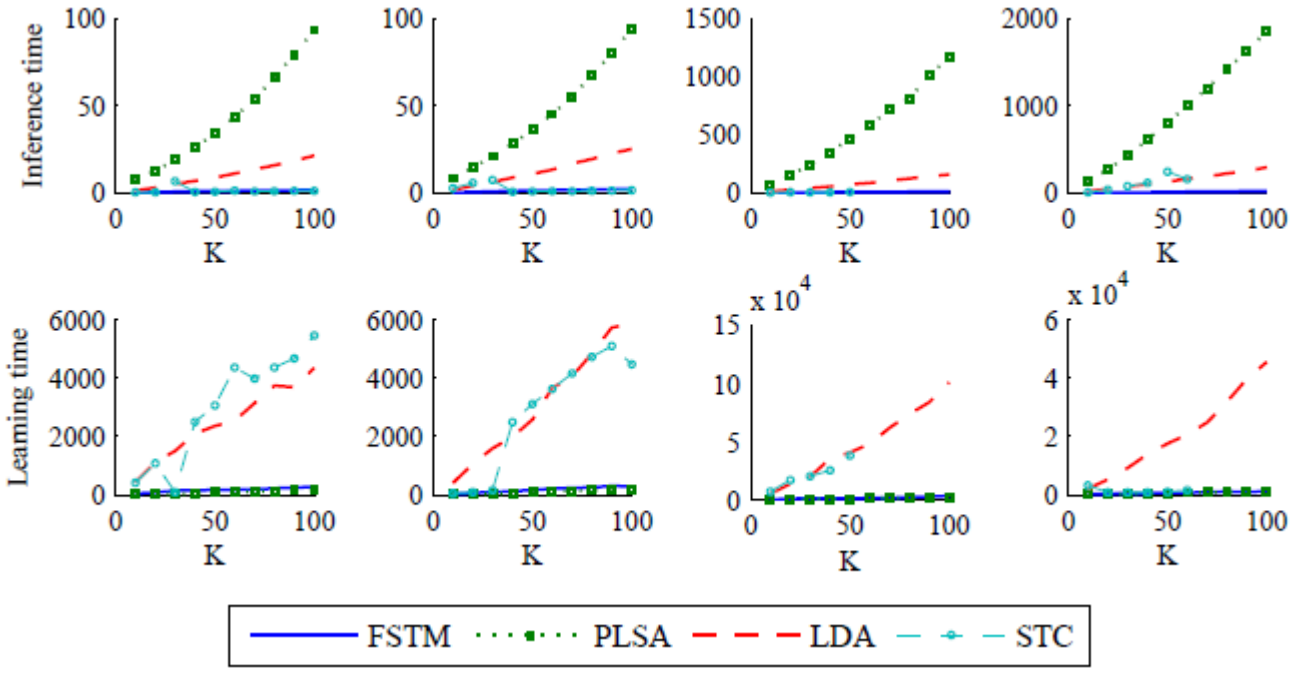
FSTM: experiments

- Sparsity: lower is better



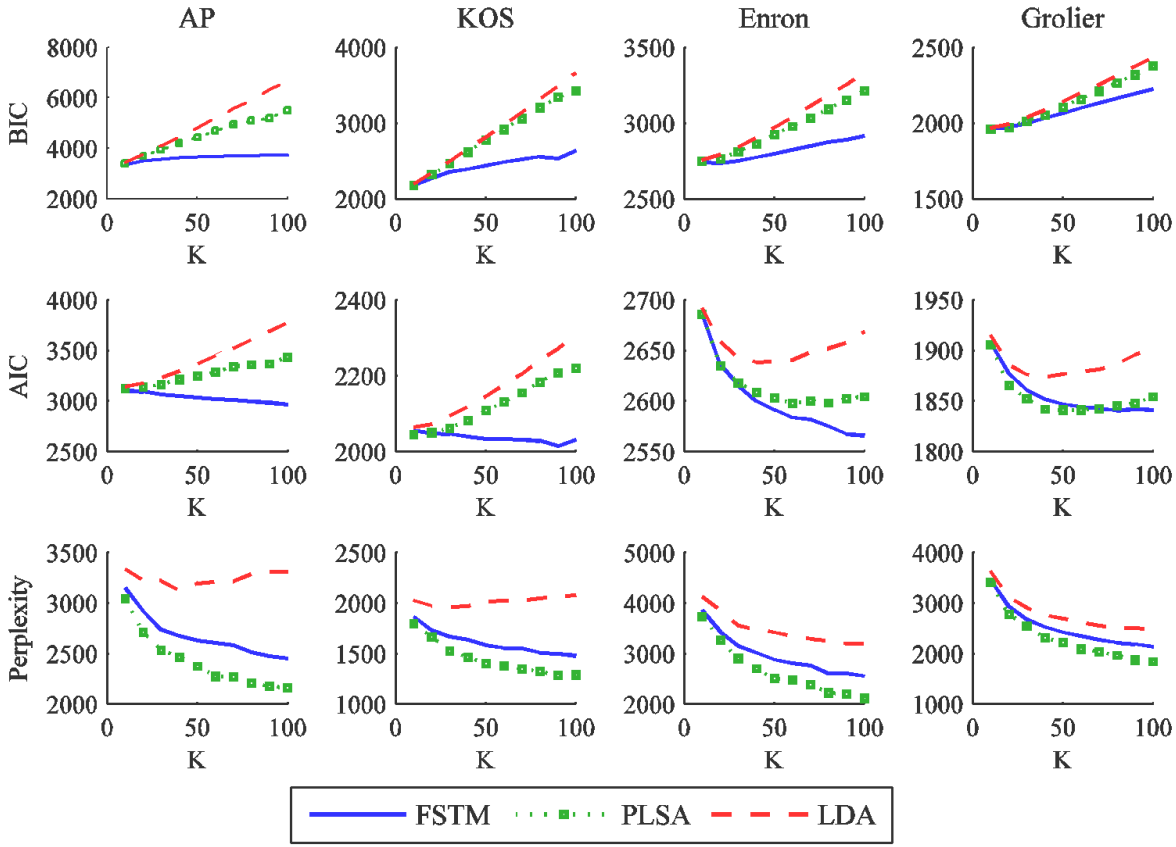
FSTM: experiments

- Speed: lower is better



FSTM: experiments

- Quality, measured by AIC, BIC, and Perplexity: lower is better

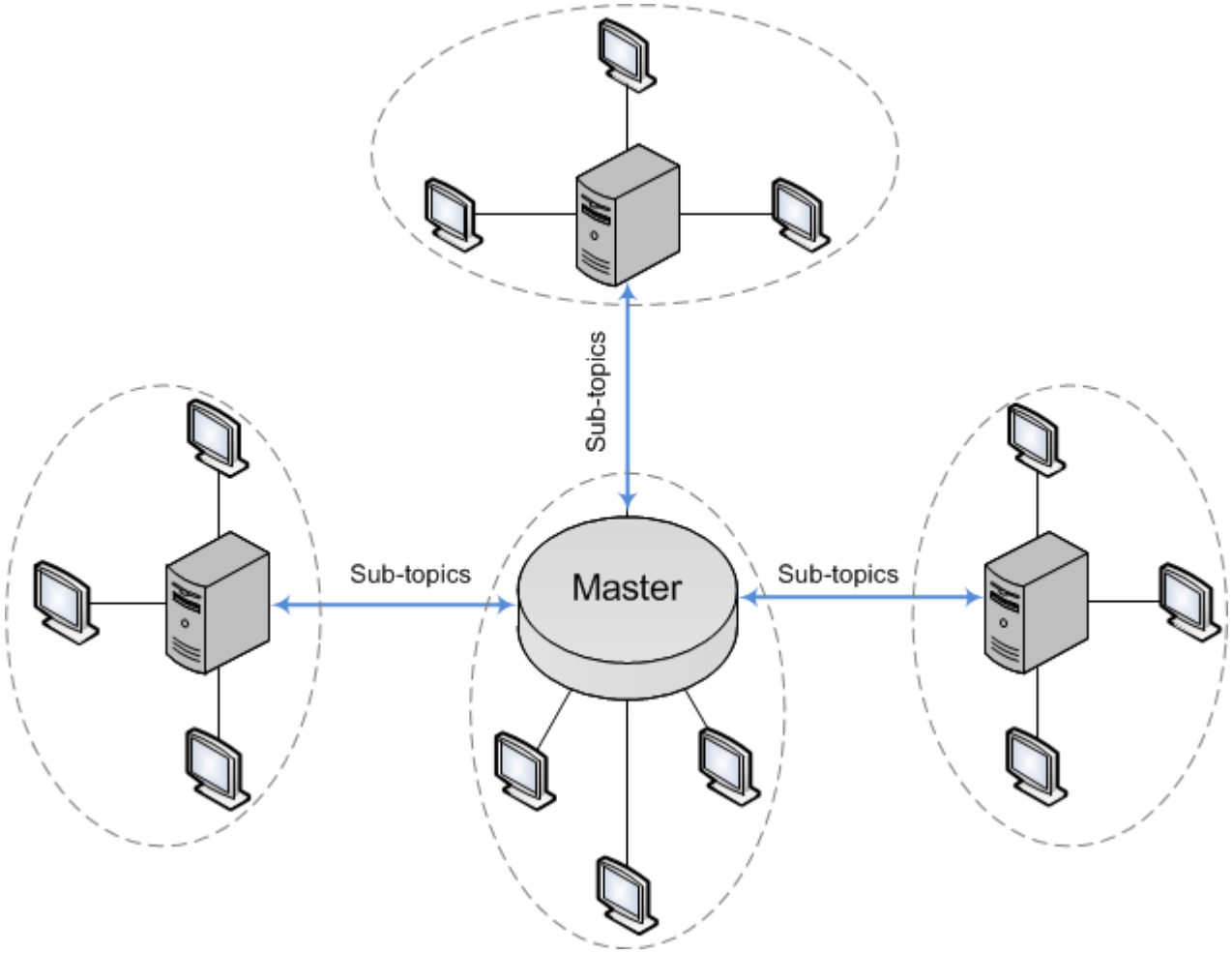


A distributed architecture

FSTM: a distributed architecture

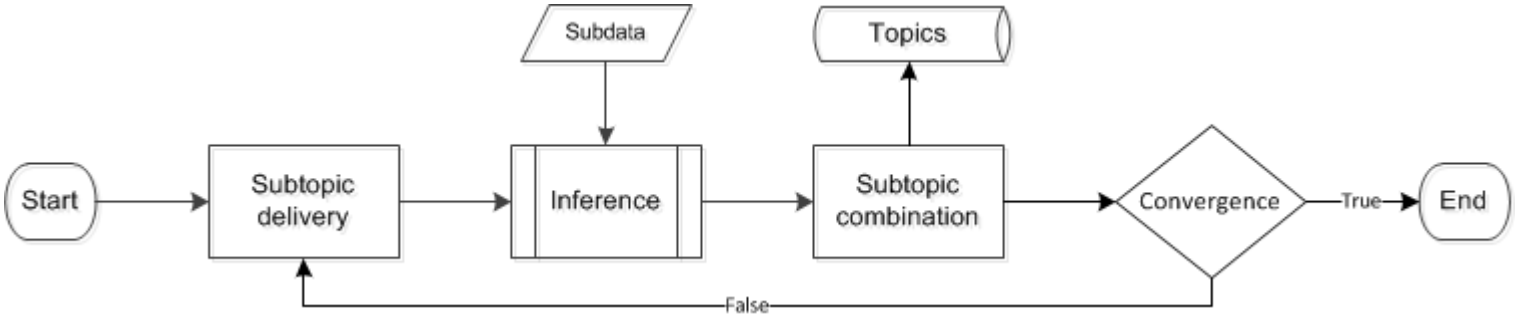
- Both parallel and distributed architectures are employed.
- CPUs are grouped into clusters.
- Each cluster has a master which will communicate with the parent master.
- Subtopics is communicated between the masters of clusters and the parent master.
- Data is distributed among the clusters before learning.

FSTM: a distributed architecture



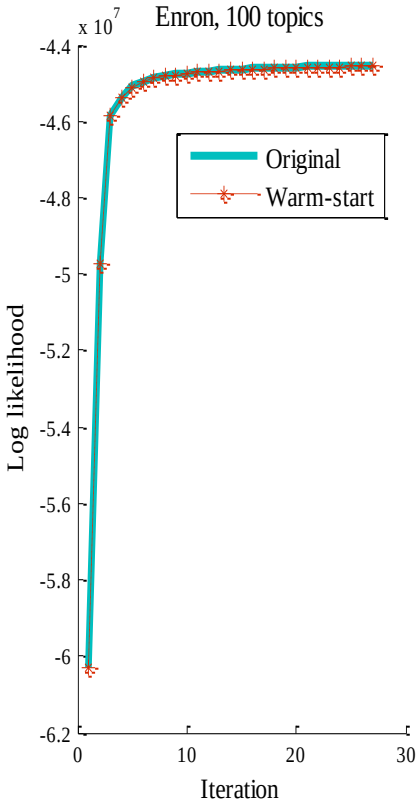
FSTM: workflow

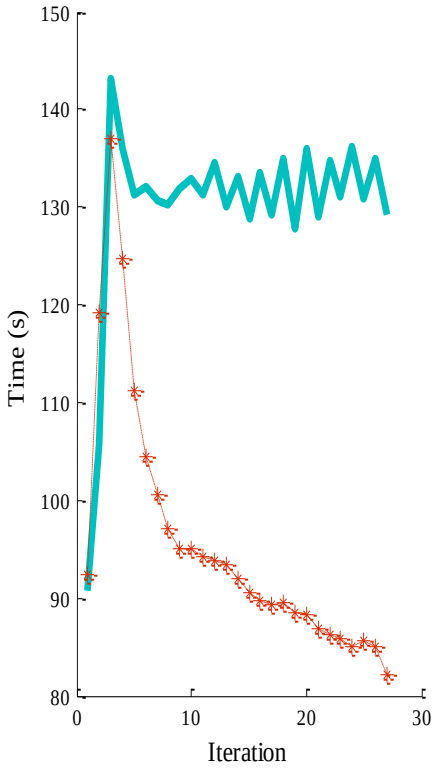
- Each cluster has its own subset of the training data.
  - Before inference, the master of a cluster retrieves necessary subtopics from the parent master.
  - After inference on data, the master send the achieved statistics of topics to the parent master.
  - The parent master constructs topics from the received statistics.
  - If not convergence, delivers topics to the children.



FSTM: accelerating inference

- Warm-start is employed.
  - For each document, the inference result in the previous EM iteration is used to guide inference in this step.
  - The most probable topics to the document is selected at the initial step of the inference algorithm.
- This helps significantly accelerate inference.
- But could lose some accuracy (empirically negligible).





FSTM: large-scale learning

- Large-scale learning
  - FSTM is implemented using OpenMP.
  - A machine with 128 CPUs is used, each with 2.9GHz, grouped into 32 clusters each having 4 CPUs
  - Webspam is selected, which has 350,000 documents, with more than 16 millions of dimensions.
  - Number of topics: 2000
  - #Latent variables for dense models: > 33 billions

http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary/webspam_wc_normalized_trigram.svm.bz2

FSTM: large-scale learning

- Learning result:

#Topics	1000	2000
Time per EM iteration	28 minutes	65 minutes
#EM iterations to reach convergence	17	16
Topic sparsity (compared with dense models)	0.0165 (60 times smaller)	0.0114 (87 times smaller)
Document sparsity (compared with dense models)	0.0054 (185 times smaller)	0.0028 (357 times smaller)
Storage for the new representation (compared with the original corpus)	31.5 Mb (757 times smaller)	33.2 Mb (718 times smaller)

FSTM: large-scale learning

- Quality of the inferred representation
  - Meaningless if the inferred representation loses too much information.
  - Classification was used to check meaningfulness.
- FSTM inferred new representation of Webspam.
- Then we used Liblinear to do classification on it.

Data	#documents	#dimensions	Storage	Best known Accuracy	Classified by	Repetitions
Original Webspam	350,000	16,609,143	23.3 Gb	99.15%	BMD [Yu et al. 2012]	1
Represented by FSTM						
1000 topics	350,000	1000	31.5 Mb	98.877%	FSTM + Liblinear	5
2000 topics	350,000	2000	33.2 Mb	99.146%	FSTM + Liblinear	5

Conclusion

Conclusion and open problems

- Conclusion:
  - Achieved our targets.
  - Provides a model for very large-scale modeling.
  - The proposed model is well-qualified and competitive.
- Open problems:
  - Online learning to sequentially deal with huge data.
  - Incorporating prior knowledge.
  - Making ease the inference for existing/future models.
  - Learning/inference when the model cannot fit in memory.

References

- Asuncion, A.U., Smyth, P., Welling, M.: Asynchronous distributed estimation of topic models for document analysis. *Statistical Methodology* 8(1) (2011) 3-17.
- Bekkerman et al. (2012), *"Scaling up Machine Learning"*, Cambridge press.
- Blei D. M., Ng A. Y., Jordan M. I. (2003), "Latent Dirichlet allocation", *Journal of Machine Learning Research*, 3, pp. 993–1022.
- Clarkson, K.L.: Coresets, sparse greedy approximation, and the frank-wolfe algorithm. *ACM Trans. Algorithms* 6 (2010) 63:1-63:30
- Hofmann T. (2001), "Unsupervised Learning by Probabilistic Latent Semantic Analysis", *Machine Learning*, 42(1), pp. 177-196.
- Hoffman, M.D., Blei, D.M., Bach, F. (2010) ,"Online learning for latent dirichlet allocation", In: *Advances in Neural Information Processing Systems*. Volume 23. (2010), 856-864
- Landauer, T.K. and Dumais, S.T. (1997), "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge", *Psychological Review*, vol. 104(2), 211-240.
- Smola, A., Narayanamurthy, S. (2010) ,"An architecture for parallel topic models", *Proceedings of the VLDB Endowment* 3(1-2) , 703-710
- Sontag, D., Roy, D.M.: Complexity of inference in latent dirichlet allocation. In: *Advances in Neural Information Processing Systems (NIPS)*. (2011)
- Wang, Q., Xu, J., Li, H., Craswell, N.: Regularized latent semantic indexing. In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '11, ACM (2011) 685-694
- Zhu, J., Xing, E.P.: Sparse topical coding. In: *UAI*. (2011)
- Yu, H.F., Hsieh, C.J., Chang, K.W., Lin, C.J.: Large linear classification when data cannot fit in memory. *ACM Trans. Knowl. Discov. Data* 5(4) (2012) 23:1-23:23

Thank you for patient listening.

Address for more discussion:
Khoat Than,
School of Knowledge Science,
Japan Advanced Institute of Science and Technology,
Email: khoat@jaist.ac.jp