An effective framework for supervised dimension reduction

Khoat Than[a,*], Tu Bao Ho[b,c], Duy Khuong Nguyen[b,c]

[a] *Hanoi University of Science and Technology,*
1 Dai Co Viet road, Hanoi, Vietnam
[b] *Japan Advanced Institute of Science and Technology,*
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan
[c] *University of Engineering and Technology, Vietnam National University, Hanoi*

Abstract

We consider supervised dimension reduction (SDR) for problems with discrete inputs. Existing methods are computationally expensive, and often do not take the local structure of data into consideration when searching for a low-dimensional space. In this paper, we propose a novel framework for SDR with the aims that it can inherit scalability of existing unsupervised methods, and that it can exploit well label information and local structure of data when searching for a new space. The way we encode local information in this framework ensures three effects: preserving inner-class local structure, widening inter-class margin, and reducing possible overlap between classes. These effects are vital for success in practice. Such an encoding helps our framework succeed even in cases that data points reside in a nonlinear manifold, for which existing methods fail.
The framework is general and flexible so that it can be easily adapted to various unsupervised topic models. We then adapt our framework to three unsupervised models which results in three methods for SDR. Extensive experiments on 10 practical domains demonstrate that our framework can yield scalable and qualitative methods for SDR. In particular, one of the adapted methods can perform consistently better than the state-of-the-art method for SDR while enjoying 30-450 times faster speed.

Keywords: supervised dimension reduction, topic models, scalability, local structure, manifold learning

1. Introduction

In supervised dimension reduction (SDR), we are asked to find a low-dimensional space which preserves the predictive information of the response variable. Projection on that space should keep the discrimination property of data in the original space. While there is a rich body of researches on SDR, our primary focus in this paper is on developing methods for discrete data. At least three reasons motivate our study: (1) current state-of-the-art methods for continuous data are really computationally expensive [1, 2, 3], and hence can only deal with data of small size and low dimensions; (2) meanwhile, there are excellent developments which can work well on discrete data of huge size [4, 5] and extremely high dimensions [6], but are unexploited for supervised problems; (3) further, continuous data can be easily discretized to avoid sensitivity and to effectively exploit certain algorithms for discrete data [7].

Topic modeling is a potential approach to dimension reduction. Recent advances in this new area can deal well with huge data of very high dimensions [4, 6, 5]. However, due to their unsupervised nature, they do not exploit supervised information. Furthermore, because the local structure of data in the original space is not considered appropriately, the new space is not guaranteed to preserve the discrimination property and proximity between instances. These limitations make unsupervised topic models unappealing to supervised dimension reduction.

Investigation of local structure in topic modeling have been initiated by some previous researches [8, 9, 10]. These are basically extensions of *probabilistic latent semantic analysis* (PLSA) by Hofmann [11],

*Corresponding author. This work was done when the author was at JAIST.

Email addresses: khoattq@soict.hust.edu.vn (Khoat Than), bao@jaist.ac.jp (Tu Bao Ho), khuongnd@jaist.ac.jp (Duy Khuong Nguyen)

which take local structure of data into account. Local structures are derived from nearest neighbors, and are often encoded in a graph. Those structures are then incorporated into the likelihood function when learning PLSA. Such an incorporation of local structures often results in learning algorithms of very high complexity. For instances, the complexity of each iteration of the learning algorithms by Wu et al. [8] and Huh and Fienberg [9] is *quadratic* in the size M of the training data; and that by Cai et al. [10] is *triple* in M because of requiring a matrix inversion. Hence these developments, even though often being shown to work well, are very limited when the data size is large.

Some topic models [12, 13, 14] for supervised problems can do simultaneously two nice jobs. One job is derivation of a meaningful space which is often known as *"topical space"*. The other is that supervised information is explicitly utilized by max-margin approach [14] or likelihood maximization [12]. Nonetheless, there are two common limitations of existing supervised topic models. First, the local structure of data is not taken into account. Such an ignorance can hurt the discrimination property in the new space. Second, current learning methods for those supervised models are often very expensive, which is problematic with large data of high dimensions.

In this paper, we approach to SDR in a novel way. Instead of developing new supervised models, we propose the *two-phase* framework which can inherit scalability of recent advances for unsupervised topic models, and can exploit label information and local structure of the training data. The main idea behind the framework is that we first learn an unsupervised topic model to find an initial topical space; we next project documents on that space exploiting label information and local structure, and then reconstruct the final space. To this end, we employ the Frank-Wolfe algorithm [15] for fast doing projection/inference.

The way of encoding local information in this framework ensures three effects: preserving inner-class local structure, widening inter-class margin, and reducing possible overlap between classes. These effects are vital for success in practice. We find that such encoding helps our framework succeed even in cases that data points reside in a nonlinear manifold, for which existing methods might fail. Further, we find that ignoring either label information (as in [9]) or manifold structure (as in [14, 16]) can significantly worsen quality of the low-dimensional space. This finding complements a recent theoretical study [17] which shows that, for some semi-supervised problems, using manifold information would definitely improve quality.

Our framework for SDR is general and flexible so that it can be easily adapted to various unsupervised topic models. To provide some evidences, we adapt our framework to three models: *probabilistic latent semantic analysis* (PLSA) by Hofmann [11], *latent Dirichlet allocation* (LDA) by Blei et al. [18], and *fully sparse topic models* (FSTM) by Than and Ho [6]. The resulting methods for SDR are respectively denoted as PLSA^c, LDA^c, and FSTM^c. Extensive experiments on 10 practical domains show that PLSA^c, LDA^c, and FSTM^c can perform substantially better than their unsupervised counterparts.[1] They perform comparably or better than existing methods that base either on max-margin principle such as MedLDA [14], or on manifold regularization without using labels such as DTM [9]. Further, PLSA^c and FSTM^c consumes significantly less time than MedLDA and DTM to learn good low-dimensional spaces. These results suggest that the two-phase framework provides a competitive approach to supervised dimension reduction.

Organization: in the next section, we describe briefly some notations, the Frank-Wolfe algorithm, and related unsupervised topic models. We present the proposed framework for SDR in Section 3. We also discuss in Section 4 the reasons why label information and local structure of data can be exploited well to result in good methods for SDR. Empirical evaluation is presented in Section 5. Finally, we discuss some open problems and conclusions in the last section.

2. Background

Consider a corpus $\mathcal{D} = \{\boldsymbol{d}_1, ..., \boldsymbol{d}_M\}$ consisting of M documents which are composed from a vocabulary of V terms. Each document $\boldsymbol{d}$ is represented as a vector of term frequencies, i.e. $\boldsymbol{d} = (d_1, ..., d_V) \in \mathbb{R}^V$, where d_j is the number of occurrences of term j in $\boldsymbol{d}$. Let $\{y_1, ..., y_M\}$ be the class labels assigned to

[1] Note that due to being dimension reduction methods, PLSA, LDA, FSTM, PLSA^c, LDA^c, and FSTM^c themselves cannot directly do classification. Hence we use SVM with a linear kernel for doing classification tasks on the low-dimensional spaces. Performance for comparison is the accuracy of classification.

those documents, respectively. The task of *supervised dimension reduction* (SDR) is to find a new space of K dimensions which preserves the predictiveness of the response/label variable Y. Loosely speaking, predictiveness preservation requires that projection of data points onto the new space should preserve separation (discrimination) between classes in the original space, and that proximity between data points is maintained. Once the new space is determined, we can work with projections in that low-dimensional space instead of the high-dimensional one.

2.1. Unsupervised topic models

Probabilistic topic models often assume that a corpus is composed of K topics, and each document is a mixture of those topics. Example models includes PLSA [11], LDA [18], and FSTM [6]. Under a model, each document has another latent representation, known as *topic proportion*, in the K-dimensional space. Hence topic models play a role as dimension reduction if $K < V$. Learning a low-dimensional space is equivalent to learning the topics of a model. Once such a space is learned, new documents can be projected onto that space via *inference*. Next, we describe briefly how to learn and to do inference for three models.

2.1.1. PLSA

Let $\theta_{dk} = P(z_k|\boldsymbol{d})$ be the probability that topic k appears in document $\boldsymbol{d}$, and $\beta_{kj} = P(w_j|z_k)$ be the probability that term j contributes to topic k. These definitions basically imply that $\sum_{k=1}^{K} \theta_{dk} = 1$ for each $\boldsymbol{d}$, and $\sum_{j=1}^{V} \beta_{kj} = 1$ for each topic k. The PLSA model assumes that document $\boldsymbol{d}$ is a mixture of K topics, and $P(z_k|\boldsymbol{d})$ is the proportion that topic k contributes to $\boldsymbol{d}$. Hence the probability of term j appearing in $\boldsymbol{d}$ is $P(w_j|\boldsymbol{d}) = \sum_{k=1}^{K} P(w_j|z_k)P(z_k|\boldsymbol{d}) = \sum_{k=1}^{K} \theta_{dk}\beta_{kj}$. Learning PLSA is to learn the topics $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_K)$. Inference of document $\boldsymbol{d}$ is to find $\boldsymbol{\theta}_d = (\theta_{d1}, ..., \theta_{dK})$.

For learning, we use the EM algorithm to maximize the likelihood of the training data:

$$\text{E-step:} \quad P(z_k|\boldsymbol{d}, w_j) = \frac{P(w_j|z_k)P(z_k|\boldsymbol{d})}{\sum_{l=1}^{K} P(w_j|z_l)P(z_l|\boldsymbol{d})}, \quad (1)$$

$$\text{M-step:} \quad \theta_{dk} = P(z_k|\boldsymbol{d}) \propto \sum_{v=1}^{V} d_v P(z_k|\boldsymbol{d}, w_v), \quad (2)$$

$$\beta_{kj} = P(w_j|z_k) \propto \sum_{\boldsymbol{d}\in\mathcal{D}} d_j P(z_k|\boldsymbol{d}, w_j) \quad (3)$$

Inference in PLSA is not explicitly derived. Hofmann [11] proposed an adaptation from learning: keeping topics fixed, iteratively do the steps (1) and (2) until convergence. This algorithm is called *folding-in*.

2.1.2. LDA

Blei et al. [18] proposed LDA as a Bayesian version of PLSA. In LDA, the topic proportions are assumed to follow a Dirichlet distribution. The same assumption is endowed over topics $\boldsymbol{\beta}$. Learning and inference in LDA are much more involved than those of PLSA. Each document $\boldsymbol{d}$ is independently inferred by the variational method with the following updates:

$$\phi_{djk} \propto \beta_{kw_j} \exp \Psi(\gamma_{dk}), \quad (4)$$

$$\gamma_{dk} = \alpha + \sum_{d_j>0} \phi_{djk}, \quad (5)$$

where ϕ_{djk} is the probability that topic i generates the j^{th} word w_j of $\boldsymbol{d}$; γ_d is the variational parameters; Ψ is the digamma function; α is the parameter of the Dirichlet prior over $\boldsymbol{\theta}_d$.

Learning LDA is done by iterating the following two steps until convergence. The E-step does inference for each document. The M-step maximizes the likelihood of data w.r.t $\boldsymbol{\beta}$ by the following update:

$$\beta_{kj} \propto \sum_{\boldsymbol{d}\in\mathcal{D}} d_j \phi_{djk}. \quad (6)$$

2.1.3. FSTM

FSTM is a simplified variant of PLSA and LDA. It is the result of removing the endowment of Dirichlet distributions in LDA, and is a variant of PLSA when removing the observed variable associated with each document. Though being a simplified variant, FSTM has many interesting properties including fast inference and learning algorithms, and ability to infer sparse topic proportions for documents. Inference is done by the Frank-Wolfe algorithm which is provably fast. Learning of topics is simply a multiplication of the new and old representations of the training data.

$$\beta_{kj} \propto \sum_{\boldsymbol{d}\in\mathcal{D}} d_j \theta_{dk}. \quad (7)$$

2.2. The Frank-Wolfe algorithm for inference

Inference is an integral part of probabilistic topic models. The main task of inference for a given document is to infer the topic proportion that maximizes a certain objective function. The most common objectives are likelihood and posterior probability. Most algorithms for inference are model-specific and are

Algorithm 1 Frank-Wolfe

Input: concave objective function $f(\boldsymbol{\theta})$.
Output: $\boldsymbol{\theta}$ that maximizes $f(\boldsymbol{\theta})$ over Δ.
Pick as $\boldsymbol{\theta}_0$ the vertex of Δ with largest f value.
for $\ell = 0, ..., \infty$ **do**
  $i' := \arg\max_i \nabla f(\boldsymbol{\theta}_\ell)_i$;
  $\alpha' := \arg\max_{\alpha \in [0,1]} f(\alpha \boldsymbol{e}_{i'} + (1-\alpha)\boldsymbol{\theta}_\ell)$;
  $\boldsymbol{\theta}_{\ell+1} := \alpha' \boldsymbol{e}_{i'} + (1-\alpha')\boldsymbol{\theta}_\ell$.
end for

nontrivial to be adapted to other models. A recent study by Than and Ho [19] reveals that there exists a highly scalable algorithm for sparse inference that can be easily adapted to various models. That algorithm is very flexible so that an adaptation is simply a choice of an appropriate objective function. Details are presented in Algorithm 1, in which $\Delta = \{\boldsymbol{x} \in \mathbb{R}^K : ||\boldsymbol{x}||_1 = 1, \boldsymbol{x} \geq 0\}$ denotes the unit simplex in the K-dimensional space. The following theorem indicates some important properties.

Theorem 1. *[15] Let f be a continuously differentiable, concave function over Δ, and denote C_f be the largest constant so that $f(\alpha \boldsymbol{x}' + (1-\alpha)\boldsymbol{x}) \geq f(\boldsymbol{x}) + \alpha(\boldsymbol{x}' - \boldsymbol{x})^t \nabla f(\boldsymbol{x}) - \alpha^2 C_f, \forall \boldsymbol{x}, \boldsymbol{x}' \in \Delta, \alpha \in [0,1]$. After ℓ iterations, the Frank-Wolfe algorithm finds a point $\boldsymbol{\theta}_\ell$ on an $(\ell+1)-$dimensional face of Δ such that $\max_{\boldsymbol{\theta} \in \Delta} f(\boldsymbol{\theta}) - f(\boldsymbol{\theta}_\ell) \leq 4C_f/(\ell+3)$.*

3. The two-phase framework for supervised dimension reduction

Existing methods for SDR often try to find directly a low-dimensional space (called *discriminative space*) that preserves separation of the data classes in the original space. Those are one-phase algorithms as depicted in Figure 1.

We propose a novel framework which consists of two phases. Loosely speaking, the first phase tries to find an initial topical space, while the second phase tries to utilize label information and local structure of the training data to find the discriminative space. The first phase can be done by employing an unsupervised topic model [6, 4], and hence inherits its scalability. Label information and local structure in the form of neighborhood will be used to guide projection of documents onto the initial space, so that inner-class local structure is preserved, inter-class margin is widen, and possible overlap between classes is reduced. As a consequence, the discrimination property is not only preserved, but likely made better in the final space.

Algorithm 2 Two-phase framework for SDR

Phase 1: learn an unsupervised model to get K topics $\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_K$. Let $\mathfrak{A} = span\{\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_K\}$ be the initial space.
Phase 2: (finding discriminative space)

(2.1) for each class c, select a set S_c of topics which are potentially discriminative for c.

(2.2) for each document $\boldsymbol{d}$, select a set N_d of its nearest neighbors which are in the same class as $\boldsymbol{d}$.

(2.3) infer new representation $\boldsymbol{\theta}^*_d$ for each document $\boldsymbol{d}$ in class c using the Frank-Wolfe algorithm with the objective function

$$f(\boldsymbol{\theta}) = \lambda L(\widehat{\boldsymbol{d}}) + \frac{1-\lambda}{|N_d|} \sum_{\boldsymbol{d}' \in N_d} L(\widehat{\boldsymbol{d}'}) + R \sum_{j \in S_c} \sin\theta_j,$$

where $L(\widehat{\boldsymbol{d}})$ is the log likelihood of document $\widehat{\boldsymbol{d}} = \boldsymbol{d}/||\boldsymbol{d}||_1$; $\lambda \in [0,1]$ and R are nonnegative constants.

(2.4) compute new topics $\boldsymbol{\beta}^*_1, ..., \boldsymbol{\beta}^*_K$ from all $\boldsymbol{d}$ and $\boldsymbol{\theta}^*_d$. Finally, $\mathfrak{B} = span\{\boldsymbol{\beta}^*_1, ..., \boldsymbol{\beta}^*_K\}$ is the discriminative space.

Note that we do not have to design entirely a learning algorithm as for existing approaches, but instead do one further inference phase for the training documents. Details of the two-phase framework are presented in Algorithm 2. Each step from (2.1) to (2.4) will be detailed in the next subsections.

3.1. Selection of discriminative topics

It is natural to assume that the documents in a class are talking about some specific topics which are little mentioned in other classes. Those topics are discriminative in the sense that they help us distinguish classes. Unsupervised models do not consider discrimination when learning topics, hence offer no explicit mechanism to see discriminative topics.

We use the following idea to find potentially discriminative topics: a topic that is discriminative for class c if its contribution to c is significantly greater than to other classes. The contribution of topic k to

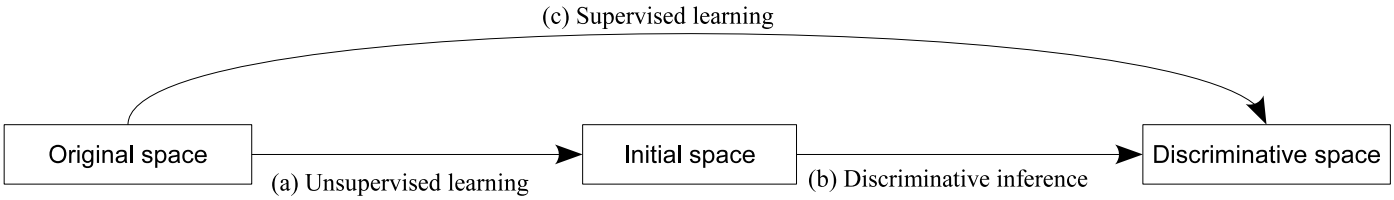

Figure 1: Sketch of approaches for SDR. Existing methods for SDR directly find the discriminative space, which is known as supervised learning (c). Our framework consists of two separate phases: (a) first find an initial space in an unsupervised manner; then (b) utilize label information and local structure of data to derive the final space.

class c is approximated by

$$T_{ck} \propto \sum_{\boldsymbol{d} \in \mathcal{D}_c} \theta_{dk},$$

where $\mathcal{D}_c$ is the set of training documents in class c, $\boldsymbol{\theta}_d$ is the topic proportion of document $\boldsymbol{d}$ which had been inferred previously from an unsupervised model. We assume that topic k is discriminative for class c if

$$\frac{T_{ck}}{\min\{T_{1k}, ..., T_{Ck}\}} \geq \epsilon, \tag{8}$$

where C is the total number of classes, ϵ is a constant which is not smaller than 1.

ϵ can be interpreted as the boundary to differentiate which classes a topic is discriminative for. For intuition, considering the problem with 2 classes, condition (8) says that topic k is discriminative for class 1 if its contribution to k is at least ϵ times the contribution to class 2. If ϵ is too large, there is a possibility that a certain class might not have any discriminative topic. On the other hand, a too small value of ϵ may yield non-discriminative topics. Therefore, a suitable choice of ϵ is necessary. In our experiments we find that $\epsilon = 1.5$ is appropriate and reasonable. We further constraint $T_{ck} \geq median\{T_{1k}, ..., T_{Ck}\}$ to avoid the topic that contributes equally to most classes.

3.2. Selection of nearest neighbors

The use of nearest neighbors in Machine Learning have been investigated by various researches [8, 9, 10]. Existing investigations often measure proximity of data points by cosine or Euclidean distances. In contrast, we use the Kullback-Leibler divergence (KL). The reason comes from the fact that projection/inference of a document onto the topical space inherently uses KL divergence.[2] Hence the use of KL divergence to find nearest neighbors is more reasonable than that of cosine or Euclidean distances in topic modeling. Note that we find neighbors for a given document $\boldsymbol{d}$ within the class containing $\boldsymbol{d}$, i.e., neighbors are local and within-class. We use $KL(\boldsymbol{d}||\boldsymbol{d}')$ to measure proximity from $\boldsymbol{d}'$ to $\boldsymbol{d}$.

3.3. Inference for each document

Let S_c be the set of potentially discriminative topics of class c, and N_d be the set of nearest neighbors of a given document $\boldsymbol{d}$ which belongs to c. We next do inference for $\boldsymbol{d}$ again to find the new representation $\boldsymbol{\theta}^*_d$. At this stage, inference is not done by existing method of the unsupervised model in consideration. Instead, the Frank-Wolfe algorithm is employed, with the following objective function to be maximized:

$$f(\boldsymbol{\theta}) = \lambda L(\widehat{\boldsymbol{d}}) + (1-\lambda)\frac{1}{|N_d|}\sum_{\boldsymbol{d}' \in N_d} L(\widehat{\boldsymbol{d}'}) + R\sum_{j \in S_c} \sin\theta_j, \tag{9}$$

where $L(\widehat{\boldsymbol{d}}) = \sum_{j=1}^{V} \hat{d}_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}$ is the log likelihood of document $\widehat{\boldsymbol{d}} = \boldsymbol{d}/||\boldsymbol{d}||_1$; $\lambda \in [0, 1]$ and R are nonnegative constants.

It is worthwhile making some observations about implication of this choice of objective:

- First, note that function $\sin(x)$ monotonically increases as x increases from 0 to 1. Therefore, the last term of (9) implies that we are promoting contributions of the topics in S_c to document $\boldsymbol{d}$. In other words, since $\boldsymbol{d}$ belongs to class c and S_c contains the topics which are potentially discriminative for c, the projection of $\boldsymbol{d}$ onto the topical

[2] For instance, consider inference of document $\boldsymbol{d}$ by maximum likelihood. Inference is the problem $\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} L(\widehat{\boldsymbol{d}}) = \arg\max_{\boldsymbol{\theta}} \sum_{j=1}^{V} \hat{d}_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}$, where $\hat{d}_j = d_j/||\boldsymbol{d}||_1$. Denoting $\boldsymbol{x} = \boldsymbol{\beta}\boldsymbol{\theta}$, the inference problem is reduced to $\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} \sum_{j=1}^{V} \hat{d}_j \log x_j = \arg\min_{\boldsymbol{x}} KL(\widehat{\boldsymbol{d}}||\boldsymbol{x})$. This implies inference of a document inherently uses KL divergence.

space should remain large contributions of the topics of S_c. Increasing the constant R implies heavier promotion of contributions of the topics in S_c.

- Second, the term $\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d} L(\widehat{\boldsymbol{d}'})$ implies that the local neighborhood plays a role when projecting $\boldsymbol{d}$. The smaller the constant λ, the more heavily the neighborhood plays. Hence, this additional term ensures that the local structure of data in the original space should not be violated in the new space.

- In practice, we do not have to store all neighbors of a document in order to do inference. Indeed, storing the mean $\boldsymbol{v} = \frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d}\widehat{\boldsymbol{d}'}$ is sufficient, since $\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d} L(\widehat{\boldsymbol{d}'}) = \frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d}\sum_{j=1}^{V}\hat{d}'_j \log\sum_{k=1}^{K}\theta_k\beta_{kj} = \sum_{j=1}^{V}\left(\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d}\hat{d}'_j\right)\log\sum_{k=1}^{K}\theta_k\beta_{kj}$.

- It is easy to verify that $f(\boldsymbol{\theta})$ is continuously differentiable and concave over the unit simplex Δ if $\boldsymbol{\beta} > 0$. As a result, the Frank-Wolfe algorithm can be seamlessly employed for doing inference. Theorem 1 guarantees that inference of each document is very fast and the inference error is provably good.

3.4. Computing new topics

One of the most involved parts in our framework is to construct the final space from the old and new representations of documents. PLSA and LDA do not provide a direct way to compute topics from $\boldsymbol{d}$ and $\boldsymbol{\theta}^*_d$, while FSTM provides a natural one. We use (7) to find the discriminative space for FSTM,

$$\text{FSTM:} \qquad \beta^*_{kj} \propto \sum_{\boldsymbol{d}\in\mathcal{D}} d_j\theta^*_{dk}; \tag{10}$$

and use the following adaptations to compute topics for PLSA and LDA:

$$\text{PLSA:} \quad \tilde{P}(z_k|\boldsymbol{d},w_j) \propto \theta^*_{dk}\beta_{kj}, \tag{11}$$

$$\beta^*_{kj} \propto \sum_{\boldsymbol{d}\in\mathcal{D}} d_j\tilde{P}(z_k|\boldsymbol{d},w_j); \tag{12}$$

$$\text{LDA:} \qquad \phi^*_{djk} \propto \beta_{kw_j}\exp\Psi(\theta^*_{dk}), \tag{13}$$

$$\beta^*_{kj} \propto \sum_{\boldsymbol{d}\in\mathcal{D}} d_j\phi^*_{djk}. \tag{14}$$

Note that we use the topics of the unsupervised models which had been learned previously in order to find the final topics. As a consequence, this usage provides a chance for unsupervised topics to affect discrimination of the final space. In contrast, using (10) to compute topics for FSTM does not encounter this drawback, and hence can inherit discrimination of $\boldsymbol{\theta}^*$. For LDA, the new representation $\boldsymbol{\theta}^*_d$ is temporarily considered to be variational parameter in place of $\boldsymbol{\gamma}_d$ in (4), and is smoothed by a very small constant to make sure the existence of $\Psi(\theta^*_{dk})$. Other adaptations are possible to find $\boldsymbol{\beta}^*$, nonetheless, we observe that our proposed adaptation is very reasonable. The reason is that computation of $\boldsymbol{\beta}^*$ uses as little information from unsupervised models as possible, whereas inheriting label information and local structure encoded in $\boldsymbol{\theta}^*$, to reconstruct the final space $\mathfrak{B} = span\{\boldsymbol{\beta}^*_1, ..., \boldsymbol{\beta}^*_K\}$. This reason is further supported by extensive experiments as discussed later.

4. Why is the framework good?

We next elucidate the main reasons for why our proposed framework is reasonable and can result in a good method for SDR. In our observations, the most important reason comes from the choice of the objective (9) for inference. Inference with that objective plays three crucial roles to preserve or make better the discrimination property of data in the topical space.

4.1. Preserving inner-class local structure

The first role is to preserve inner-class local structure of data. This is a result of using the additional term $\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d} L(\widehat{\boldsymbol{d}'})$. Remember that projection of document $\boldsymbol{d}$ onto the unit simplex Δ is in fact a search for the point $\boldsymbol{\theta}_d \in \Delta$ that is closest to $\boldsymbol{d}$ in a certain sense.[3] Hence if $\boldsymbol{d}'$ is close to $\boldsymbol{d}$, it is natural to expect that $\boldsymbol{d}'$ is close to $\boldsymbol{\theta}_d$. To respect this nature and to keep the discrimination property, projecting a document should take its local neighborhood into account. As one can realize, the part $\lambda L(\widehat{\boldsymbol{d}}) + (1-\lambda)\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d} L(\widehat{\boldsymbol{d}'})$ in the objective (9) serves well our needs. This part interplays goodness-of-fit and neighborhood preservation. Increasing λ means goodness-of-fit $L(\widehat{\boldsymbol{d}})$ can be improved, but local structure around $\boldsymbol{d}$ is prone to be broken in the low-dimensional space. Decreasing λ implies better preservation of local structure. Figure 2 demonstrates sharply these two extremes, $\lambda = 1$ for (b), and

[3] More precisely, the vector $\sum_k \theta_{dk}\boldsymbol{\beta}_k$ is closest to $\boldsymbol{d}$ in terms of KL divergence.

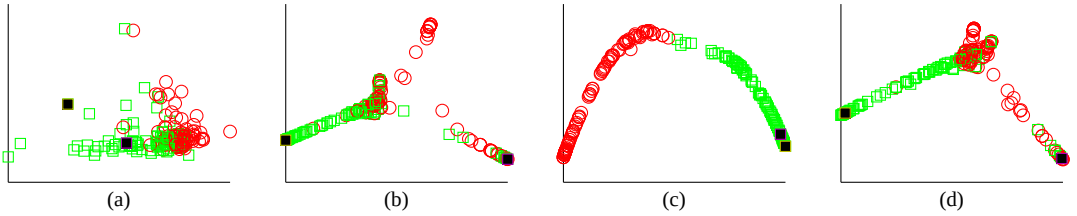

Figure 2: Laplacian embedding in 2D space. (a) data in the original space, (b) unsupervised projection, (c) projection when neighborhood is taken into account, (d) projection when topics are promoted. These projections onto the 60-dimensional space were done by FSTM and experimented on 20Newsgroups. The two black squares are documents in the same class.

$\lambda = 0.1$ for (c). Projection by unsupervised models ($\lambda = 1$) often results in pretty overlapping classes in the topical space, whereas exploitation of local structure significantly helps us separate classes.

Since nearest neighbors N_d are selected within-class only, doing projection for $\boldsymbol{d}$ in step (2.3) is not intervened by documents from outside classes. Hence within-class local structure would be better preserved.

4.2. Widening the inter-class margin

The second role is to widen the inter-class margin, owing to the term $R\sum_{j\in S_c}\sin(\theta_j)$. As noted before, function $\sin(x)$ is monotonically increasing for $x \in [0, 1]$. It implies that the term $R\sum_{j\in S_c}\sin(\theta_j)$ promotes contributions of the topics in S_c when projecting document $\boldsymbol{d}$. In other words, the projection of $\boldsymbol{d}$ is encouraged to be close to the topics which are potentially discriminative for class c. Hence projection of class c is preferred to distributing around the discriminative topics of c. Increasing the constant R implies forcing projections to distribute more densely around the discriminative topics, and therefore making classes farther from each other. Figure 2(d) illustrates the benefit of this second role.

4.3. Reducing overlap between classes

The third role is to reduce overlap between classes, owing to the term $\lambda L(\widehat{\boldsymbol{d}}) + (1-\lambda)\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d} L(\widehat{\boldsymbol{d}'})$ in the objective function (9). This is a very crucial role that helps the two-phase framework works effectively. Explanation for this role needs some insights into inference of $\boldsymbol{\theta}$.

In step (2.3), we have to do inference for the training documents. Let $\boldsymbol{u} = \lambda\widehat{\boldsymbol{d}} + (1-\lambda)\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d}\widehat{\boldsymbol{d}'}$ be the convex combination of $\boldsymbol{d}$ and its within-class neighbors.[4] Note that

$$\begin{aligned}
& \lambda L(\widehat{\boldsymbol{d}}) + (1-\lambda)\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d} L(\widehat{\boldsymbol{d}'}) \\
= \;& \lambda\sum_{j=1}^{V}\widehat{d}_j\log\sum_{k=1}^{K}\theta_k\beta_{kj} \\
& +(1-\lambda)\frac{1}{|N_d|}\sum_{\boldsymbol{d}'\in N_d}\sum_{j=1}^{V}\widehat{d'}_j\log\sum_{k=1}^{K}\theta_k\beta_{kj} \\
= \;& \sum_{j=1}^{V}\left(\lambda\widehat{d}_j + \frac{1-\lambda}{|N_d|}\sum_{\boldsymbol{d}'\in N_d}\widehat{d'}_j\right)\log\sum_{k=1}^{K}\theta_k\beta_{kj} \\
= \;& L(\boldsymbol{u}).
\end{aligned}$$

Hence, in fact we do inference for $\boldsymbol{u}$ by maximizing $f(\boldsymbol{\theta}) = L(\boldsymbol{u}) + R\sum_{j\in S_c}\sin(\theta_j)$. It implies that we actually work with $\boldsymbol{u}$ in the U-space as depicted in Figure 3.

Those observations suggest that instead of working with the original documents in the document space, we do work with $\{\boldsymbol{u}_1, ..., \boldsymbol{u}_M\}$ in the U-space. Figure 3 shows that the classes in the U-space is often less overlapping than those in the document space. Further, the overlapping can sometimes be removed. Hence working in the U-space would be probably more effective than in the document space, in the sense of supervised dimension reduction.

5. Evaluation

This section is dedicated to investigating effectiveness and efficiency of our framework in practice. We investigate three methods, PLSA^c, LDA^c, and

[4] More precisely, $\boldsymbol{u}$ is the convex combination of those documents in ℓ_1-normalized forms, since by notation $\widehat{\boldsymbol{d}} = \boldsymbol{d}/||\boldsymbol{d}||_1$.

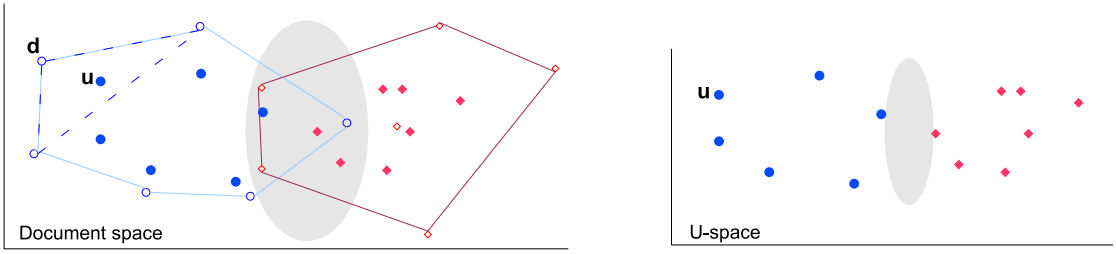

Figure 3: The effect of reducing overlap between classes. In Phase 2 (discriminative inference), inferring $\boldsymbol{d}$ is reduced to inferring $\boldsymbol{u}$ which is the convex combination of $\boldsymbol{d}$ and its within-class neighbors. This means we are working in the U-space instead of the document space. Note that the classes in the U-space are often much less overlapping than those in the document space.

FSTM^c, which are the results of adapting the two-phase framework to unsupervised topic models including PLSA [11], LDA [18], and FSTM [6], respectively.

Methods for comparison:

- MedLDA: the baseline which bases on max-margin principle [14], but ignores manifold structure when learning.[5]
- DTM: the baseline which uses manifold regularization, but ignores labels [9].
- PLSA^c, LDA^c, and FSTM^c: the results of adapting our framework to three unsupervised models.
- PLSA, LDA, and FSTM: three unsupervised methods associated with three models.[6]

Data for comparison: We use 10 benchmark datasets for investigation which span over various domains including news in LA Times, biological articles, spam emails. Table 1 shows some information about those data.[7]

Settings: In our experiments, we used the same criteria for topic models: relative improvement of the log likelihood (or objective function) is less than 10^{-4} for learning, and 10^{-6} for inference; at most 1000 iterations are allowed to do inference; and at most 100 iterations for learning a model/space. The same criterion was used to do inference by the Frank-Wolfe algorithm in Phase 2 of our framework.

Table 1: Statistics of data for experiments

Data	Training size	Testing size	Dimensions	Classes
LA1s	2566	638	13196	6
LA2s	2462	613	12433	6
News3s	7663	1895	26833	44
OH0	805	198	3183	10
OH5	739	179	3013	10
OH10	842	208	3239	10
OH15	735	178	3101	10
OHscal	8934	2228	11466	10
20Newsgroups	15935	3993	62061	20
Emailspam	3461	866	38729	2

MedLDA is a supervised topic model and is trained by minimizing a hinge loss. We used the best setting as studied by [14] for some other parameters in MedLDA: cost parameter $\ell = 32$, and 10-fold cross-validation for finding the best regularization constant $C \in \{25, 29, 33, 37, 41, 45, 49, 53, 57, 61\}$. These settings were chosen to avoid possibly biased comparison.

For DTM, we used 20 neighbors for each data instance when constructing neighborhood graphs. We also tried to use 5 and 10, but found that fewer neighbors did not improve quality significantly. We set $\lambda = 1000$ meaning that local structure plays a heavy role when learning a space. Further, because DTM itself does not provide any method for doing projection of new data onto a discriminative space. Hence we implemented the Frank-Wolfe algorithm which does projection for new data by maximizing their likelihood.

For the two-phase framework, we set $N_d = 20, \lambda = 0.1, R = 1000$. This setting basically says that local neighborhood plays a heavy role when projecting

[5] MedLDA was retrieved from `www.ml-thu.net/~jun/code/MedLDAc/medlda.zip`

[6] LDA was taken from `www.cs.princeton.edu/~blei/lda-c/`
FSTM was taken from `www.jaist.ac.jp/~s1060203/codes/fstm/`
PLSA was written by ourselves with the best effort.

[7] 20Newsgroups was taken from `www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`. Emailspam was taken from `csmining.org/index.php/spam-email-datasets-.html`. Other datasets were retrieved from the UCI repository.

documents, and that classes are very encouraged to be far from each other in the topical space.

It is worth noting that the two-phase framework plays the main role in searching for the discriminative space $\mathfrak{B}$. Hence, other works aftermath such as projection for new documents are done by the inference methods of the associated unsupervised models. For instance, FSTM^c works as follows: we first train FSTM in an unsupervised manner to get an initial space $\mathfrak{A}$; we next do Phase 2 of Algorithm 2 to find the discriminative space $\mathfrak{B}$; projection of documents onto $\mathfrak{B}$ then is done by the inference method of FSTM which does not need label information. LDA^c and PLSA^c work in the same manner.

5.1. Quality and meaning of the discriminative spaces

Separation of classes in low-dimensional spaces is our first concern. A good method for SDR should preserve inter-class separation in the original space. Figure 4 depicts an illustration of how good different methods are, for 60 topics (dimensions). One can observe that projection by FSTM can maintain separation between classes to some extent. Nonetheless, because of ignoring label information, a large number of documents have been projected onto incorrect classes. On the contrary, FSTM^c and MedLDA exploited seriously label information for projection, and hence the classes in the topical space separate very cleanly. The good preservation of class separation by MedLDA is mainly due to training by max margin principle. Each iteration of the algorithm tries to widen the expected margin between classes. FSTM^c can separate the classes well owing to the fact that projecting documents has seriously taken local neighborhood into account, which very likely keeps inter-class separation of the original data. Furthermore, it also tries to widen the margin and reduces overlap between classes as discussed in Section 4.

Figure 5 demonstrates failures of MedLDA and DTM, while FSTM^c succeeded. For the two datasets, MedLDA learned a space in which the classes are heavily mixed. These behaviors seem strange to MedLDA, as it follows the max-margin approach which is widely-known able to learn good classifiers. In our observations, at least two reasons that may cause such failures: first, documents of LA1s (and LA2s) seem to reside on a nonlinear manifold (like a cone) so that no hyperplane can separate well one class from the rest. This may worsen performance of a classifier with an inappropriate kernel. Second, quality of the topical space learned by MedLDA is heavily affected by the quality of the classifiers which are learned at each iteration of MedLDA. When a classifier is bad (e.g., due to inappropriate use of kernels), it might worsen learning a new topical space. This situation might have happened with MedLDA on LA1s and LA2s.

DTM seems to do better than MedLDA owing to the use of local structure when learning. Nonetheless, separation of the classes in the new space learned by DTM is unclear. The main reason may be that DTM did not use label information of the training data when searching for a low-dimensional space. In contrast, the two-phase framework seriously took both local structure and label information into account. The way it uses label can reduce overlap between classes as demonstrated in Figure 5. While the classes are much overlapping in the original space, they are more cleanly separated in the discriminative space found by FSTM^c.

Meaning of the discriminative spaces is demonstrated in Table 2. It presents contribution (in terms of probability) of the most probable topic to a specific class.[8] As one can observe easily, the content of each class is reflected well by a specific topic. The probability that a class assigns to its major topic is often very high compared to other topics. The major topics in two different classes are often have different meanings. Those observations suggests that the low-dimensional spaces learned by our framework are meaningful, and each dimension (topic) reflects well the meaning of a specific class. This would be beneficial for the purpose of exploration in practical applications.

5.2. Classification quality

We next use classification as a means to quantify the goodness of the considered methods. The main role of methods for SDR is to find a low-dimensional space so that projection of data onto that space preserves or even makes better the discrimination property of data in the original space. In other words, predictiveness of the response variable is preserved or improved. Classification is a good way to see such preservation or improvement.

[8] Probability of topic k in class C is approximated by $P(z_k|C) \propto \sum_{\boldsymbol{d}\in C} \theta_{dk}$, where $\boldsymbol{\theta}_d$ is the projection of document $\boldsymbol{d}$ onto the final space.

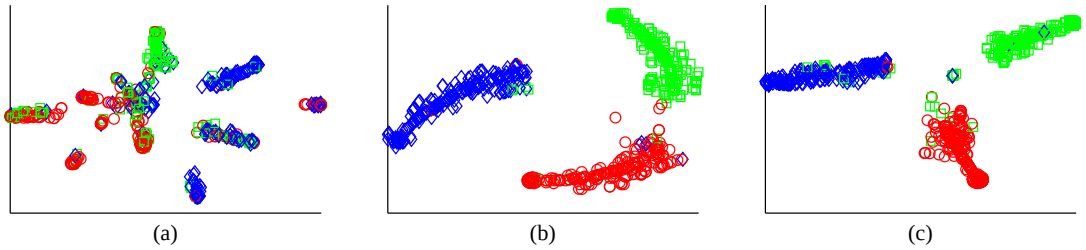

Figure 4: Projection of three classes of 20newsgroups onto the topical space by (a) FSTM, (b) FSTM^c, and (c) MedLDA. FSTM did not provide a good projection in the sense of class separation, since label information was ignored. FSTM^c and MedLDA actually found good discriminative topical spaces, and provided a good separation of classes. (These embeddings were done with t-SNE [20]. Points of the same shape (color) are in the same class.)

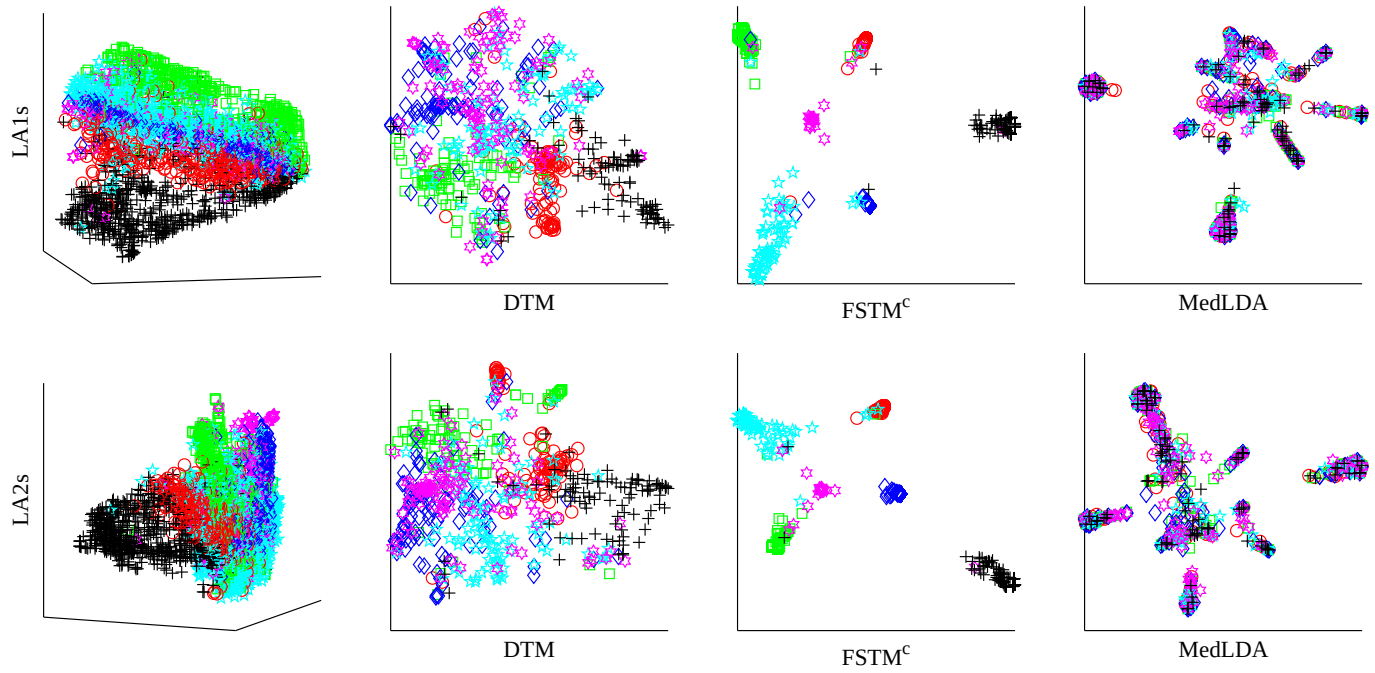

Figure 5: Failures of MedLDA and DTM when data reside on a nonlinear manifold. FSTM^c performed well so that the classes in the low-dimensional spaces were separated clearly. (These embeddings were done with t-SNE [20].)

Table 2: Meaning of the discriminative space which was learned by FSTM^c with 60 topics, from OH5. For each row, the first column shows the class label, the second column shows the topic that has highest probability in the class, and the last column shows the probability. Each topic is represented by some of the top terms. As one can observe, each topic represents well the meaning of the associated class.

Class name	**Topic** that has highest probability in class	**Probability**
Anticoagulants	anticoagul, patient, valve, embol, stroke, therapi, treatment, risk, thromboembol	0.931771
Audiometry	hear, patient, auditori, ear, test, loss, cochlear, respons, threshold, brainstem	0.958996
Child-Development	infant, children, development, age, motor, birth, develop, preterm, outcom, care	0.871983
Graft-Survival	graft, transplant, patient, surviv, donor, allograft, cell, reject, flap, recipi	0.646190
Microsomes	microsom, activ, protein, bind, cytochrom, liver, alpha, metabol, membran	0.940836
Neck	patient, cervic, node, head, injuri, complic, dissect, lymph, metastasi	0.919655
Nitrogen	nitrogen, protein, dai, nutrition, excretion, energi, balanc, patient, increas	0.896074
Phospholipids	phospholipid, acid, membran, fatti, lipid, protein, antiphospholipid, oil, cholesterol	0.875619
Radiation-Dosage	radiat, dose, dosimetri, patient, irradi, film, risk, exposur, estim	0.899836
Solutions	solution, patient, sodium, pressur, glucos, studi, concentr, effect, glycin	0.941912

For each method, we projected the training and testing data ($\boldsymbol{d}$) onto the topical space, and then used the associated projections ($\boldsymbol{\theta}$) as inputs for multi-class SVM [21] to do classification.[9] MedLDA does not need to be followed by SVM since it can do classification itself. Varying the number of topics, the results are presented in Figure 6.

Observing Figure 6, one easily realizes that the supervised methods often performed substantially better than the unsupervised ones. This suggests that FSTM^c, LDA^c, and PLSA^c exploited well label information when searching for a topical space. FSTM^c, LDA^c, and PLSA^c performed better than MedLDA when the number of topics is relatively large (≥ 60). FSTM^c consistently achieved the best performance and sometimes reached more than 10% improvement over MedLDA. Such a better performance is mainly due to the fact that FSTM^c had taken seriously local structure of data into account whereas MedLDA did not. DTM could exploit well local structure by using manifold regularization, as it performed better than PLSA, LDA, and FSTM on many datasets. However, due to ignoring label information of the training data, DTM seems to be inferior to FSTM^c, LDA^c, and PLSA^c.

Surprisingly, DTM had lower performance than PLSA, LDA, and FSTM on three datasets (LA1s, LA2s, OHscal), even though it spent intensive time trying to preserve local structure of data. Such failures of DTM might come from the fact that the classes of LA1s (or other datasets) are much overlapping in the original space as demonstrated in Figure 5. Without using label information, construction of neighborhood graphs might be inappropriate so that it hinders DTM from separating data classes. DTM puts a heavy weight on (possibly biased) neighborhood graphs which empirically approximate local structure of data. In contrast, PLSA, LDA, and FSTM did not place any bias on the data points when learning a low-dimensional space. Hence they could perform better than DTM on LA1s, LA2s, OHscal.

There is a surprising behavior of MedLDA. Though being a supervised method, it performed comparably or even worse than unsupervised methods (PLSA, LDA, FSTM) for many datasets including LA1s, LA2s, OH10, and OHscal. In particular, MedLDA performed significantly worst for LA1s and LA2s. It seems that MedLDA lost considerable information when searching for a low-dimensional space. Such a behavior has been also observed by Halpern et al. [22]. As discussed in subsection 5.1 and depicted in Figure 5, various factors might affect performance of MedLDA and other max-margin based methods. Those factors include nonlinear nature of data manifolds, ignorance of local structure, and inappropriate use of kernels when learning a topical space.

Why FSTM^c often performs best amongst three adaptations including LDA^c and PLSA^c? This question is natural, since our adaptations for three topic models use the same framework and settings. In our observations, the key reason comes from the way of deriving the final space in Phase 2. As noted before, deriving topical spaces by (12) and (14) directly requires unsupervised topics of PLSA and LDA, respectively. Such adaptations implicitly allow some chances for unsupervised topics to have direct influence on the final topics. Hence the discrimination property may be affected heavily in the new space. On the contrary, using (10) to recompute topics for FSTM does not allow a direct involvement of unsupervised topics. Therefore, the new topics can inherit almost the discrimination property encoded in $\boldsymbol{\theta}^*$. This helps the topical spaces learned by FSTM^c being more likely discriminative than those by PLSA^c and by LDA^c. Another reason is that the inference method of FSTM is provably good [6], and is often more accurate than the variational method of LDA and folding-in of PLSA [19].

5.3. Learning time

The final measure for comparison is how quickly the methods do? We mostly concern on the methods for SDR including FSTM^c, LDA^c, PLSA^c, and MedLDA. Note that time for learning a discriminative space by FSTM^c is the time to do 2 phases of Algorithm 2 which includes time to learn an unsupervised model, FSTM. The same holds for PLSA^c and LDA^c. Figure 7 summarizes the overall time for each method. Observing the figure, we find that MedLDA and LDA^c consumed intensive time, while FSTM^c and PLSA^c did substantially more speedily. One of the main reasons for slow learning of MedLDA and LDA^c is that inference by variational methods of MedLDA and LDA is often very slow. Inference in those models requires various evaluation of Digamma and gamma functions which are expensive. Further,

[9] This classification method is included in Liblinear package which is available at `www.csie.ntu.edu.tw/~cjlin/liblinear/`

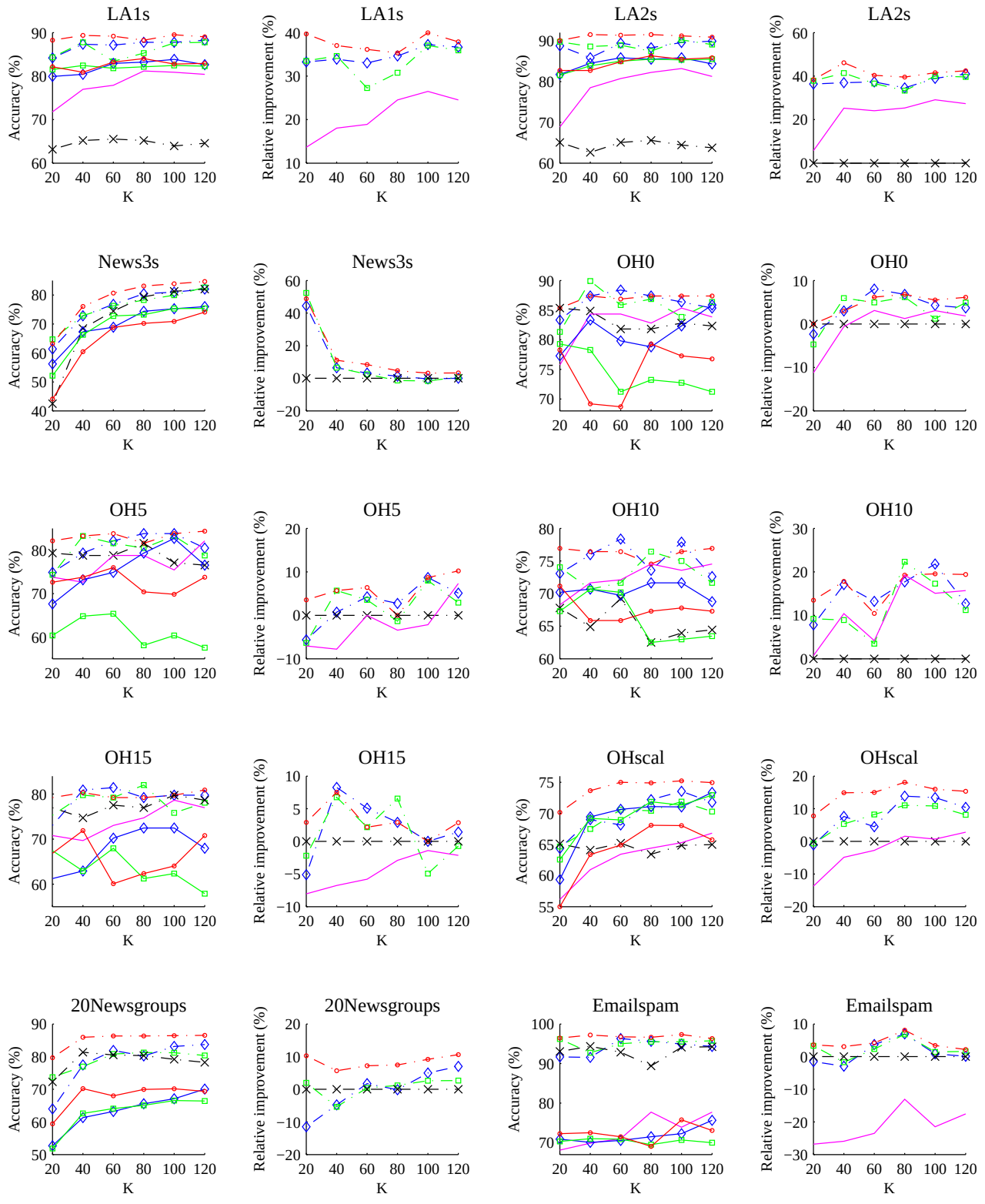

Figure 6: Accuracy of 8 methods as the number K of topics increases. Relative improvement is improvement of a method (A) over MedLDA, and is defined as $\frac{accuracy(A)-accuracy(MedLDA)}{accuracy(MedLDA)}$. DTM could not work on News3s and 20Newsgroups due to oversize memory requirement, and hence no result is reported.

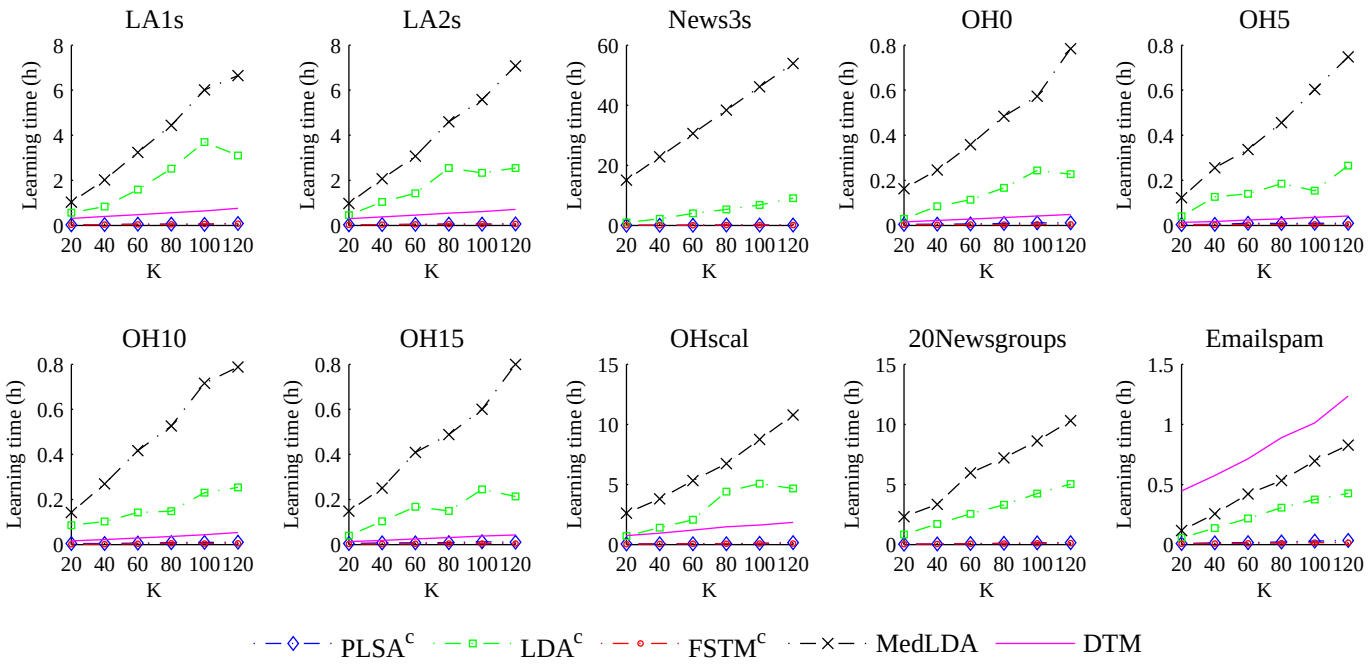

Figure 7: Necessary time to learn a discriminative space, as the number K of topics increases. FSTM^c and PLSA^c often performed substantially faster than MedLDA. As an example, for News3s and $K = 120$, MedLDA needed more than 50 hours to complete learning, whereas FSTM^c needed less than 8 minutes. (DTM is also reported to see advantages of our framework when the size of the training data is large.)

MedLDA requires a further step of learning a classifier at each EM iteration, which is empirically slow in our observations. All of these contributed to the slow learning of MedLDA and LDA^c.

In contrast, FSTM has a fast inference algorithm and requires simply a multiplication of two sparse matrices for learning topics, while PLSA has a very simple learning formulation. Hence learning in FSTM and PLSA is unsurprisingly very fast [6]. The most time consuming part of FSTM^c and PLSA^c is to search nearest neighbors for each document. A modest implementation would requires $O(V.M^2)$ arithmetic operations, where M is the data size. Such a computational complexity will be problematic when the data size is large. Nonetheless, as empirically shown in Figure 7, the overall time of FSTM^c and PLSA^c was significantly less than that of MedLDA and LDA^c. Table 3 supports further this observation. Even for 20Newsgroups and News3s of average size, learning time of FSTM^c and PLSA^c is very competitive compared with MedLDA.

Summarizing, the above investigations demonstrate that the two-phase framework can result in very competitive methods for supervised dimension reduction. Three adapted methods, FSTM^c,

Table 3: Learning time in seconds when $K = 120$. For each dataset, the first line shows the learning time and the second line shows the corresponding accuracy. The best learning time is bold, while the best accuracy is italic.

Data	PLSA^c	LDA^c	FSTM^c	MedLDA
LA1s	287.05	11,149.08	**275.78**	23,937.88
	88.24%	87.77%	*89.03%*	64.58%
LA2s	**219.39**	9,175.08	238.87	25,464.44
	89.89%	89.07%	*90.86%*	63.78%
News3s	494.72	32,566.27	**462.10**	194,055.74
	82.01%	82.59%	*84.64%*	82.01%
OH0	39.21	816.33	**16.56**	2,823.64
	85.35%	86.36%	*87.37%*	82.32%
OH5	34.08	955.77	**17.03**	2,693.26
	80.45%	78.77%	*84.36%*	76.54%
OH10	37.38	911.33	**18.81**	2,834.40
	72.60%	71.63%	*76.92%*	64.42%
OH15	38.54	769.46	**15.46**	2,877.69
	79.78%	78.09%	*80.90%*	78.65%
OHscal	584.74	16,775.75	**326.50**	38,803.13
	71.77%	70.29%	*74.96%*	64.99%
20Newsgroups	556.20	18,105.92	**415.91**	37,076.36
	83.72%	80.34%	*86.53%*	78.24%
Emailspam	124.07	1,534.90	**56.56**	2,978.18
	94.34%	95.73%	*96.31%*	94.23%

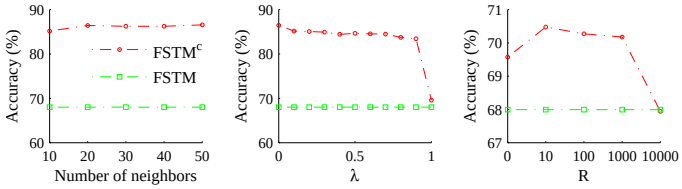

Figure 8: Impact of the parameters on the success of our framework. (left) Change the number of neighbors, while fixing $\lambda = 0.1, R = 0$. (middle) Change λ the extent of seriousness of taking local structure, while fixing $R = 0$ and using 10 neighbors for each document. (right) Change R the extent of promoting topics, while fixing $\lambda = 1$. Note that the interference of local neighborhood played a very important role, since it consistently resulted in significant improvements.

LDA^c, and PLSA^c, mostly outperform the corresponding unsupervised ones. LDA^c and PLSA^c often reached comparable performance with max-margin based methods such as MedLDA. Amongst those adaptations, FSTM^c behaves superior in both classification performance and learning speed. We observe it often does 30-450 times faster than MedLDA.

5.4. Sensitivity of parameters

There are three parameters that influence the success of our framework, including the number of nearest neighbors, λ, and R. This subsection investigates the impact of each. 20Newsgroups was selected for experiments, since it has average size which is expected to exhibit clearly and accurately what we want to see.

We varied the value of a parameter while fixed the others, and then measured the accuracy of classification. Figure 8 presents the results of these experiments. It is easy to realize that when taking local neighbors into account, the classification performance was very high and significant improvements can be achieved. We observed that very often, 25% improvement were reached when local structure was used, even with different settings of λ. These observations suggest that the use of local structure plays a very crucial role for the success of our framework. It is worth remarking that one should not use too many neighbors for each document, since performance may be worse. The reason is that using too many neighbors likely break local structure around documents. We have experienced with this phenomenon when setting 100 neighbors in Phase 2 of Algorithm 2, and got worse results.

Changing the value of R implies changing promotion of topics. In other words, we are expecting projections of documents in the new space to distribute more densely around discriminative topics, and hence making classes farther from each other. As shown in Figure 8, an increase in R often leads to better results. However, too large R can deteriorate the performance of the SDR method. The reason may be that such large R can make the term $R\sum_{j\in S_c}\sin(\theta_j)$ to overwhelm the objective (9), and thus worsen the goodness-of-fit of inference by the Frank-Wolfe algorithm. Setting $R \in [10, 1000]$ is reasonable in our observation.

6. Conclusion and discussion

We have proposed the two-phase framework for doing dimension reduction of supervised discrete data. The framework was demonstrated to exploit well label information and local structure of the training data to find a discriminative low-dimensional space. It was demonstrated to succeed in failure cases of methods which base on either max-margin principle or unsupervised manifold regularization. Generality and flexibility of our framework was evidenced by adaptation to three unsupervised topic models, resulted in PLSA^c, LDA^c, and FSTM^c for supervised dimension reduction. We showed that ignoring either label information (as in DTM) or manifold structure of data (as in MedLDA) can significantly worsen quality of the low-dimensional space. The two-phase framework can overcome existing approaches to result in efficient and effective methods for SDR. As an evidence, we observe that FSTM^c can often achieve more than 10% improvement in quality over MedLDA, and meanwhile consumes substantially less time.

The resulting methods (PLSA^c, LDA^c, and FSTM^c) are not limited to discrete data. They can work also on non-negative data, since their learning algorithms actually are very general. Hence in this work, we contributed methods for not only discrete data but also non-negative real data. The code of these methods is freely available at `www.jaist.ac.jp/~s1060203/codes/sdr/`

There is a number of possible extensions to our framework. First, one can easily modify the framework to deal with multilabel data. Second, the framework can be modified to deal with semi-supervised data. A key to these extensions is an appropriate utilization of labels to search for nearest neighbors, which is necessary for our framework. Other extensions can encode more prior knowledge into the ob-

jective function for inference. In our framework, label information and local neighborhood are encoded into the objective function and have been observed to work well. Hence, we believe that other prior knowledge can be used to derive good methods.

Of the most expensive steps in our framework is the search for nearest neighbors. By a modest implementation, it requires $O(k.V.M)$ to search k nearest neighbors for a document. Overall, finding all k nearest neighbors for all documents requires $O(k.V.M^2)$. This computational complexity will be problematic when the number of training documents is large. Hence, a significant extension would be to reduce complexity for this search. It is possible to reduce the complexity to $O(k.V.M.\log M)$ as suggested by [23]. Furthermore, because our framework use local neighborhood to guide projection of documents onto the low-dimensional space, we believe that approximation to local structure can still provide good result. However, this assumption should be studied further. A positive point of using approximation of local neighborhood is that computational complexity of a search for neighbors can be done in linear time $O(k.V.M)$ [24].

Acknowledgment

We would like to thank the two anonymous reviewers for very helpful comments. Khoat Than is supported by MEXT, Japan. T.B. Ho was partially supported by Vietnam's National Foundation for Science and Technology Development (NAFOSTED Project No. 102.99.35.09).

References

[1] M. Chen, W. Carson, M. Rodrigues, R. Calderbank, L. Carin, Communication inspired Linear Discriminant Analysis, in: Proceedings of the 29th Annual International Conference on Machine Learning, 2012.

[2] N. Parrish, M. R. Gupta, Dimensionality reduction by Local Discriminative Gaussian, in: Proceedings of the 29th Annual International Conference on Machine Learning, 2012.

[3] M. Sugiyama, Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis, The Journal of Machine Learning Research 8 (2007) 1027–1061.

[4] D. Mimno, M. D. Hoffman, D. M. Blei, Sparse stochastic inference for latent Dirichlet allocation, in: Proceedings of the 29th Annual International Conference on Machine Learning, 2012.

[5] A. Smola, S. Narayanamurthy, An architecture for parallel topic models, Proceedings of the VLDB Endowment 3 (1-2) (2010) 703–710.

[6] K. Than, T. B. Ho, Fully Sparse Topic Models, in: P. Flach, T. De Bie, N. Cristianini (Eds.), Machine Learning and Knowledge Discovery in Databases, vol. 7523 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, ISBN 978-3-642-33459-7, 490–505, URL `http://dx.doi.org/10.1007/978-3-642-33460-3-37`, 2012.

[7] Y. Yang, G. Webb, Discretization for naive-Bayes learning: managing discretization bias and variance, Machine learning 74 (1) (2009) 39–74.

[8] H. Wu, J. Bu, C. Chen, J. Zhu, L. Zhang, H. Liu, C. Wang, D. Cai, Locally discriminative topic modeling, Pattern Recognition 45 (1) (2012) 617–625.

[9] S. Huh, S. Fienberg, Discriminative topic modeling based on manifold learning, ACM Transactions on Knowledge Discovery from Data (TKDD) 5 (4) (2012) 20.

[10] D. Cai, X. Wang, X. He, Probabilistic dyadic data analysis with local and global consistency, in: Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, ACM, 105–112, URL `http://doi.acm.org/10.1145/1553374.1553388`, 2009.

[11] T. Hofmann, Unsupervised Learning by Probabilistic Latent Semantic Analysis, Machine Learning 42 (2001) 177–196, ISSN 0885-6125, URL `http://dx.doi.org/10.1023/A:1007617005950`.

[12] D. Blei, J. McAuliffe, Supervised topic models, in: Neural Information Processing Systems (NIPS), 2007.

[13] S. Lacoste-Julien, F. Sha, M. Jordan, DiscLDA: Discriminative learning for dimensionality reduction and classification, in: Advances in Neural Information Processing Systems (NIPS), vol. 21, MIT, 897–904, 2008.

[14] J. Zhu, A. Ahmed, E. P. Xing, MedLDA: maximum margin supervised topic models, The Journal of Machine Learning Research 13 (2012) 2237–2278.

[15] K. L. Clarkson, Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm, ACM Trans. Algorithms 6 (2010) 63:1–63:30, ISSN 1549-6325, doi:\bibinfo{doi}{http://doi.acm.org/10.1145/1824777.1824783}, URL `http://doi.acm.org/10.1145/1824777.1824783`.

[16] J. Zhu, N. Chen, H. Perkins, B. Zhang, Gibbs Max-Margin Topic Models with Fast Sampling Algorithms, in: ICML, vol. 28 of *Journal of Machine Learning Research: W&CP*, 124–132, 2013.

[17] P. Niyogi, Manifold Regularization and Semi-supervised Learning: Some Theoretical Analyses, Journal of Machine Learning Research 14 (2013) 1229–1250, URL `http://jmlr.org/papers/v14/niyogi13a.html`.

[18] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent Dirichlet allocation, Journal of Machine Learning Research 3 (3) (2003) 993–1022.

[19] K. Than, T. B. Ho, Managing sparsity, time, and quality of inference in topic models, Tech. Rep., 2012.

[20] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, Journal of Machine Learning Research 9 (2008) 2579–2605.

[21] S. Keerthi, S. Sundararajan, K. Chang, C. Hsieh, C. Lin, A sequential dual method for large scale multi-class linear SVMs, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 408–416, 2008.

[22] Y. Halpern, S. Horng, L. A. Nathanson, N. I. Shapiro, D. Sontag, A Comparison of Dimensionality Reduction Techniques for Unstructured Clinical Text, in: ICML 2012

Workshop on Clinical Data Analysis, 2012.

[23] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, A. Y. Wu, An optimal algorithm for approximate nearest neighbor searching fixed dimensions, Journal of the ACM 45 (6) (1998) 891–923, ISSN 0004-5411, doi:\bibinfo{doi}{10.1145/293347.293348}, URL http://doi.acm.org/10.1145/293347.293348.

[24] K. L. Clarkson, Fast algorithms for the all nearest neighbors problem, Foundations of Computer Science, IEEE Annual Symposium on 0 (1983) 226–232, ISSN 0272-5428, doi:\bibinfo{doi}{http://doi.ieeecomputersociety.org/10.1109/SFCS.1983.16}.