# Sparse Latent Semantic Analysis

**Xi Chen[1], Yanjun Qi[2], Bing Bai[2], Qihang Lin[1], Jaime G. Carbonell[1]**

1. **Machine Learning Department, Carnegie Mellon University**
2. **Machine Learning Department, NEC Lab America**

The work is done during the internship at NEC Lab America

# Background

## Vector Space Model:

$\boxed{\text{Document: } \mathbf{x} = [w_1, \ldots, w_M] \in \mathbb{R}^M}$ $M$: vocabulary size

$w_i$: normalized weight (tf-idf) of the $i$-th word

*N* Documents: $\mathbf{X} = [\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^N] \in \mathbb{R}^{N \times M}$: Document-Word matrix

## Latent Semantic Analysis:

*D* latent topics (dimensionality of the latent space)

LSA applies SVD to construct a *rank-D* approximation:

$$\mathbf{X} \approx \mathbf{U}_{N \times D} \mathbf{S}_{D \times D} (\mathbf{V}_{M \times D})^T, \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{V}^T\mathbf{V} = \mathbf{I}$$

Projection Matrix: $\mathbf{A} = \mathbf{S}^{-1}\mathbf{V}^T \in \mathbb{R}^{D \times M}$

Dimension reduction for a new document *q*: $q \in \mathbb{R}^M \Rightarrow \widehat{q} = \mathbf{A}q \in \mathbb{R}^D$

# Optimization Formulation for LSA

- Latent Semantic Analysis

$$\mathbf{X} \approx \mathbf{U}_{N\times D}\mathbf{S}_{D\times D}(\mathbf{V}_{M\times D})^T, \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \mathbf{V}^T\mathbf{V} = \mathbf{I}$$

- Relaxed Optimization Formulation:

[K. Yu et al. 05]

$$\min_{\mathbf{U},\mathbf{A}} \quad \frac{1}{2}\|\mathbf{X} - \mathbf{U}\mathbf{A}\|_F^2$$
$$\text{subject to:} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}$$

- Sparse Latent Semantic Analysis:

Add *sparsity* constraint on the project matrix $\mathbf{A}$:

$$\min_{\mathbf{U},\mathbf{A}} \quad \frac{1}{2}\|\mathbf{X} - \mathbf{U}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_1 \Longrightarrow \|\mathbf{A}\|_1 = \sum_{d=1}^{D}\sum_{j=1}^{M}|a_{dj}|$$
$$\text{subject to:} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}$$

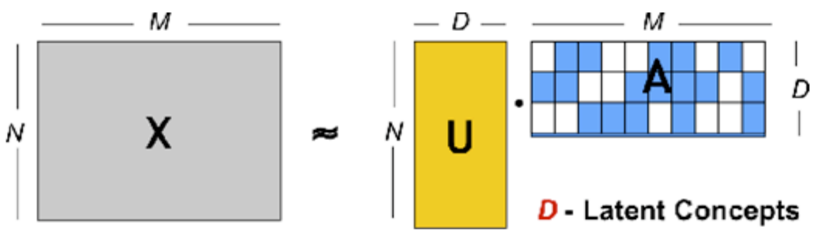
$\ell_1$-regularization

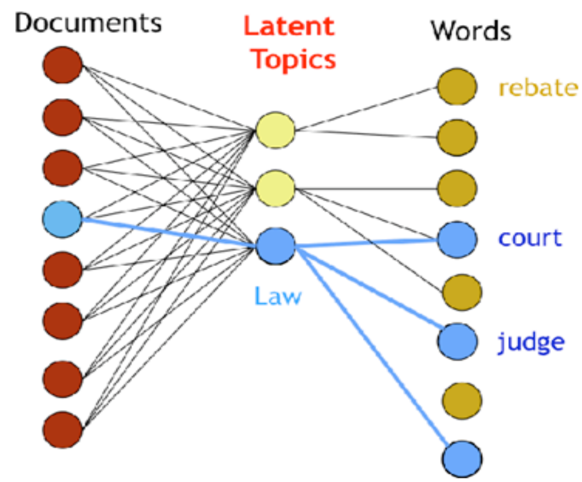
# Sparse LSA

## Sparse LSA

$$\min_{\mathbf{U},\mathbf{A}} \quad \frac{1}{2}\|\mathbf{X}-\mathbf{U}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_1$$

$$\text{subject to:} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}$$




New Document $q$: $\quad \widehat{q} = \mathbf{A}q \in \mathbb{R}^D$

Simple Projection, Computational Efficient

## Comparison to Sparse Coding

$$\min_{\mathbf{U},\mathbf{A}} \quad \frac{1}{2}\|\mathbf{X}-\mathbf{U}\mathbf{A}\|_F^2 + \lambda\|\mathbf{U}\|_1$$

$$\text{subject to:} \quad \|A_j\|_2^2 \le c, \quad j = 1,\ldots M$$

New Document $q$: $\quad \widehat{q} = \arg\min_{\widehat{q}} \frac{1}{2}\|q - \mathbf{A}^T\widehat{q}\| + \lambda\|\widehat{q}\|_1.$

Lasso Problem
More Computation Time

# Advantage of Sparse LSA

- **Better Interpretability:**

  Sparse LSA selects most relevant words for each topic ($a_{dj} \neq 0$ )

  Compact representation of topic-word relationship

- **Efficient Projection:**

  Sparse **A** $\Longrightarrow$ Efficient Projection for new documents: $\widehat{q} = \mathbf{A}q \in \mathbb{R}^D$

- **Cheap Storage:**

  Cheap storage for sparse **A**

- **Sparse Projected Documents:** sparse $\widehat{q} = \mathbf{A}q$

- **Document-Topic Relationship:** $\widehat{q}_d = 0 \Leftrightarrow \widehat{q}$ not belong to $d$-th topic

- **Advantage as compared to PCA:**

  Do not need to centralize **X** $\Longrightarrow$ destroy the sparsity of **X**

  Do not need the covariance matrix $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{M \times M}$ $\Longrightarrow$ may not fit in the memory for large vocabulary size

# Optimization Method

❖ Alternating Approach

$$\min_{\mathbf{U},\mathbf{A}} \quad \frac{1}{2}\|\mathbf{X}-\mathbf{U}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_1$$
$$\text{subject to:} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}$$

Fix **U** and optimize with respect to **A**:

$$\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{U}\mathbf{A}\| + \lambda\|\mathbf{A}\|_1$$

$$\min_{A_j} \frac{1}{2}\|X_j - \mathbf{U}A_j\|_2^2 + \lambda\|A_j\|_1; \quad j = 1,\ldots,M. \; A_j\text{: } j\text{-th column of } \mathbf{A}$$

[J. Friedman et al. 10]

*M* independent *lasso* problem : *Solved via Coordinate Descent*

Fix A and optimize with respect to **U**:

$$\min_{\mathbf{U}} \quad \frac{1}{2}\|\mathbf{X}-\mathbf{U}\mathbf{A}\|_F^2 \Leftrightarrow \mathrm{tr}(\mathbf{U}^T\mathbf{X}\mathbf{A}^T)$$
$$\text{subject to:} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}$$

Closed-form Solution:

Let $\mathbf{V} = \mathbf{X}\mathbf{A}^T$ (projected documents onto the latent space)

Perform SVD on $\mathbf{V}$: $\mathbf{V} = \mathbf{P}\Delta\mathbf{Q} \Longrightarrow \mathbf{U}^* = \mathbf{P}\mathbf{Q}$

Note: SVD on $\mathbf{V} \in \mathbb{R}^{M\times D}$ is much *cheaper* than that on $\mathbf{X} \in \mathbb{R}^{N\times M}$

# Optimization Summary

**Algorithm 1** Optimization Algorithm for Sparse LSA

**Input:** $\mathbf{X}$, dimensionality of the latent space $D$, regularization parameter $\lambda$

**Initialization:**$\mathbf{U}^0 = \begin{pmatrix} \mathbf{I}_D \\ \mathbf{0} \end{pmatrix}$,

**Iterate until convergence of U and A:**

1. Compute $\mathbf{A}$ by solving $M$ indepdent lasso problems via coordinate descent
2. Project $\mathbf{X}$ onto the latent space: $\mathbf{V} = \mathbf{X}\mathbf{A}^T$.
3. Compute the SVD of $\mathbf{V}$: $\mathbf{V} = \mathbf{P}\Delta\mathbf{Q}$ and let $\mathbf{U} = \mathbf{P}\mathbf{Q}$.

**Output:** Sparse projection matrix $\mathbf{A}$.

# Extension I : Nonnegative Sparse LSA

Constraint: $\mathbf{A} \geq 0$:

$$\begin{aligned} \min_{\mathbf{U},\mathbf{A}} \quad & \frac{1}{2}\|\mathbf{X} - \mathbf{U}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_1 \\ \text{subject to:} \quad & \mathbf{U}^T\mathbf{U} = \mathbf{I}, \quad \mathbf{A} \geq 0. \end{aligned}$$

Simulate the *probability* of the word $w_j$ given the topic $t_d$:

Normalize each row: $\widetilde{a}_{dj} = \frac{a_{dj}}{\sum_{j=1}^{M} a_{dj}} \sim \mathbb{P}(w_j|t_d)$

Optimization with respect to $\mathbf{A}$:

$$\min_{A_j \geq \mathbf{0}} f(\mathbf{A}_j) = \frac{1}{2}\|X_j - \mathbf{U}A_j\| + \lambda \sum_{d=1}^{D} a_{dj}. \quad j = 1, \ldots, M$$

Optimize via the *coordinate descent* approach:
Iterating over $d$: fix $a_{\widehat{d}j}$ for $\widehat{d} \neq d$ and optimize over $a_{dj}$

$$a_{dj}^* = \begin{cases} \frac{b_d - \lambda}{c_d} & b_d > \lambda \\ 0 & b_d \leq \lambda \end{cases},$$

$$c_d = \sum_{i=1}^N u_{id}^2, b_d = \sum_{i=1}^N u_{id}(x_{ij} - \sum_{k \neq d} u_{ik}a_{kj}).$$

# Extension II : Group Structured Sparse LSA

- Application: latent gene-function identification: determine relevant pathways (groups of genes) to a latent gene function (topic)
- Group Structured Sparse LSA:

The set of groups of input features $\mathcal{G} = \{g_1, \ldots, g_{|\mathcal{G}|}\}$ (available as *a priori*)

$$\min_{\mathbf{U},\mathbf{A}} \quad \frac{1}{2}\|\mathbf{X} - \mathbf{U}\mathbf{A}\|_F^2 + \lambda \sum_{d=1}^{D} \sum_{g \in \mathcal{G}} w_g \|\mathbf{A}_{dg}\|_2$$

$$\text{subject to:} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}.$$

Optimization with respect to $\mathbf{A}$:

Optimize via the *coordinate descent* approach:

$$\mathbf{A}^*_{dg} = \begin{cases} \frac{\mathbf{B}_{dg}(\|\mathbf{B}_{dg}\|_2 - \lambda w_g)}{C_d \|\mathbf{B}_{dg}\|_2} & \|\mathbf{B}_{dg}\|_2 > \lambda w_g \\ \mathbf{0} & \|\mathbf{B}_{dg}\|_2 \le \lambda w_g \end{cases}.$$

$$C_d = \sum_{i=1}^{N} u_{id}^2, \; (\mathbf{B}_{dg})_{j \in g} = \sum_{i=1}^{N} u_{id}(x_{ij} - \sum_{k \ne d} u_{ik} a_{kj}).$$

# Experimental Setup

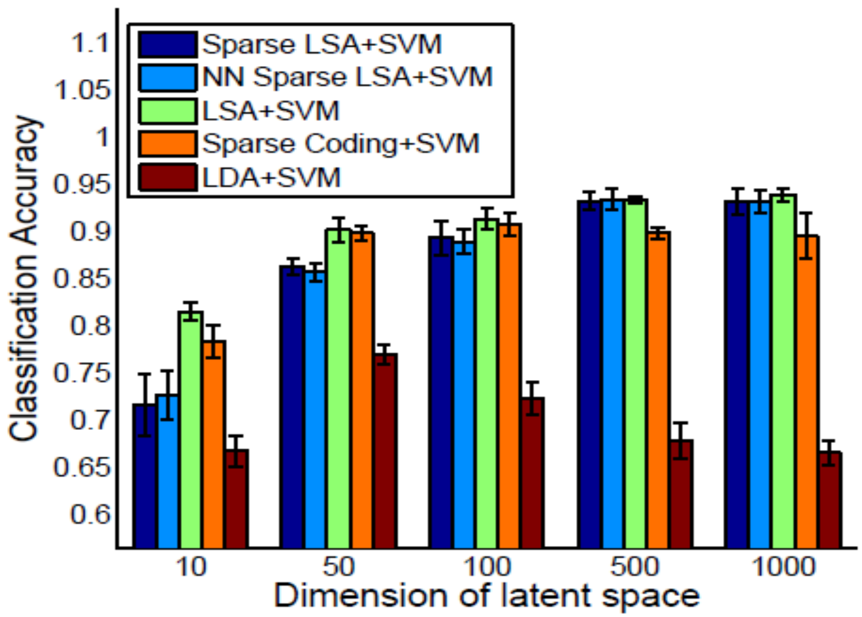
| Methods Compared |
| --- |
| Traditional LSA |
| Sparse Coding (Code from Lee et. al. 07) |
| Latent Dirichlet allocation (LDA) (Code from Blei et. al. 03) |
| Sparse LSA |
| Nonnegative Sparse LSA (NN Sparse LSA) |

| Text Classification Data | *N* (No. of Documents) | *M* ( Vocabulary Size) |
| --- | --- | --- |
| 20 news group (20NG) (alt.atheism vs talk.religion.misc) | 1,425 | 17,390 |
| RCV1 (20 classes) | 15,564 | 7,413 |

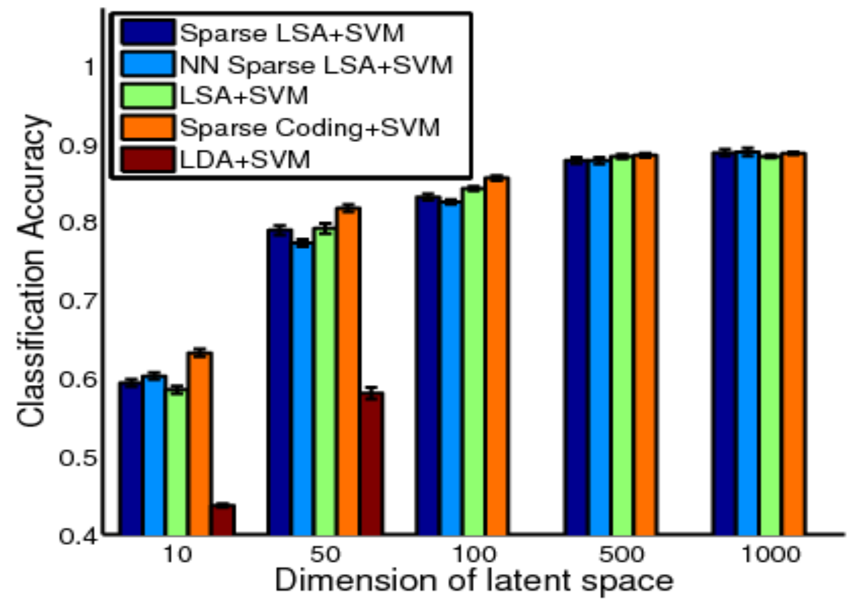
| Topic-Word Relationship Data | *N* (No. of Documents) | *M* ( Vocabulary Size) |
| --- | --- | --- |
| NIPS Proceedings from 98 to 99 | 1,714 | 13,649 |

# Text Classification Performance

20NG




RCV1




| Dimension | 10 | 50 | 100 | 500 | 1000 |
|---|---|---|---|---|---|
| Sparse LSA | 1.48 | 0.80 | 0.74 | 0.32 | 0.18 |
| NN Sparse LSA | 1.44 | 0.72 | 0.55 | 0.31 | 0.17 |
| Other Methods | 100 | 100 | 100 | 100 | 100 |

20NG: Density of **A** (%) ($\lambda$=0.05)

| Dimension | 10 | 50 | 100 | 500 | 1000 |
|---|---|---|---|---|---|
| Sparse LSA | 13.52 | 7.46 | 7.40 | 2.71 | 1.13 |
| NN Sparse LSA | 11.65 | 4.97 | 0.40 | 1.91 | 0.79 |
| Other Methods | 100 | 100 | 100 | 100 | 100 |

RCV1: Density of **A** (%) ($\lambda$=0.05)

Conclusion: For large $D$, the classification performance of *Sparse LSA* is almost the same as *LSA* but with a much more *sparse* projection matrix **A**.

# Efficiency and Storage

20NG

| | Proj. Time (ms) | Storage (MB) | Density of Proj. Doc. (%) |
|---|---|---|---|
| Sparse LSA | 0.25 (4.05E-2) | 0.6314 | 35.81 (15.39) |
| NN Sparse LSA | 0.22 (2.78E-2) | 0.6041 | 35.44 (15.17) |
| LSA | 31.6 (1.10) | 132.68 | 100 (0) |
| Sparse Coding | 1711.1 (323.9) | 132.68 | 86.94 (3.63) |

RCV1

| | Proj. Time (ms) | Storage (MB) | Density of Proj. Doc. (%) |
|---|---|---|---|
| Sparse LSA | 0.59 (7.36E-2) | 1.3374 | 55.38 (11.77) |
| NN Sparse LSA | 0.46 (6.66E-2) | 0.9537 | 46.47 (11.90) |
| LSA | 13.2 (0.78) | 113.17 | 100 (0) |
| Sparse Coding | 370.5 (23.3) | 113.17 | 83.88 (2.11) |

Conclusion: Sparse LSA or NN Sparse LSA

- Efficient projection with less time
- Less storage for the projection matrix $\mathbf{A}$
- Sparse projected documents: more efficient for subsequent retrieval tasks, e.g. ranking, text categorization, etc

$D = 1,000, \lambda = 0.05$ Table entry : mean (std)

# Topic–Word Relationship

NIPS from 1988 to 1999

## Nonnegative Sparse LSA

| Topic 1 | Topic 2 | Topic 3 | Topic 4 |
|---|---|---|---|
| network | learning | network | model |
| neural | reinforcement | learning | data |
| networks | algorithm | data | models |
| system | function | neural | parameters |
| neurons | rule | training | mixture |
| neuron | control | set | likelihood |
| input | learn | function | distribution |
| output | weight | model | gaussian |
| time | action | input | em |
| systems | policy | networks | variables |

| Topic 5 | Topic 6 | Topic 7 |
|---|---|---|
| function | input | image |
| functions | output | images |
| approximation | inputs | recognition |
| linear | chip | visual |
| basis | analog | object |
| threshold | circuit | system |
| theorem | signal | feature |
| loss | current | figure |
| time | action | input |
| systems | policy | networks |

## LDA

| Topic 1 | Topic 2 | Topic 3 | Topic 4 |
|---|---|---|---|
| learning | figure | algorithm | single |
| data | model | method | general |
| model | output | networks | sets |
| training | neurons | process | time |
| information | vector | learning | maximum |
| number | networks | input | paper |
| algorithm | state | based | rates |
| performance | layer | function | features |
| linear | system | error | estimated |
| input | order | parameter | neural |

| Topic 5 | Topic 6 | Topic 7 |
|---|---|---|
| rate | algorithms | function |
| unit | set | neural |
| data | problem | hidden |
| time | weight | networks |
| estimation | temporal | recognition |
| node | prior | output |
| set | obtain | visual |
| input | parameter | noise |
| neural | neural | parameters |
| properties | simulated | references |

Conclusion: The topics learned by *NN Sparse LSA* are discriminative while the topics learned by *LDA* are all closely related to *neural network*.

# Thank You !