Statistical Debugging with Latent Topic Models

David Andrzejewski, Anne Mulhern, Ben Liblit, Xiaojin Zhu

Department of Computer Sciences
University of Wisconsin–Madison

European Conference on Machine Learning, 2007

Outline

1. Statistical Debugging
2. Our Approach
3. Our Results

A problem has been detected and windows has been shut down to prevent damage
to your computer.

The problem seems to be caused by the following file: SPCMDCON.SYS

PAGE_FAULT_IN_NONPAGED_AREA

If this is the first time you've seen this Stop error screen,
restart your computer. If this screen appears again, follow
these steps:

Check to make sure any new hardware or software is properly installed.
If this is a new installation, ask your hardware or software manufacturer
for any windows updates you might need.

If problems continue, disable or remove any newly installed hardware
or software. Disable BIOS memory options such as caching or shadowing.
If you need to use Safe Mode to remove or disable components, restart
your computer, press F8 to select Advanced Startup Options, and then
select Safe Mode.

Technical information:

*** STOP: 0x00000050 (0xFD3094C2,0x00000001,0xFBFE7617,0x00000000)

*** SPCMDCON.SYS - Address FBFE7617 base at FBFE5000, DateStamp 3d6dd67c

Oh no!

- It didn't crash last time...
- What's going on here?

```
A problem has been detected and Windows has been shut down to prevent damage
to your computer.

The problem seems to                                      MDCON.SYS

PAGE_FAULT_IN_NONPAGE

If this is the first                                         een,
restart your computer. If this screen appears again, follow
these steps:

Check to make sure any new hardware or software is properly installed.
If this is a new installation, ask your hardware or software manufacturer
for any Windows updates you might need.

If problems continue, disable or remove any newly installed hardware
or software. Disable BIOS memory options such as caching or shadowing.
If you need to use Safe Mode to remove or disable components, restart
your computer, press F8 to select Advanced Startup Options, and then
select Safe Mode.

Technical information:

*** STOP: 0x00000050 (0xFD3094C2,0x00000001,0xFBFE7617,0x00000000)


***  SPCMDCON.SYS - Address FBFE7617 base at FBFE5000, DateStamp 3d6dd67c
```

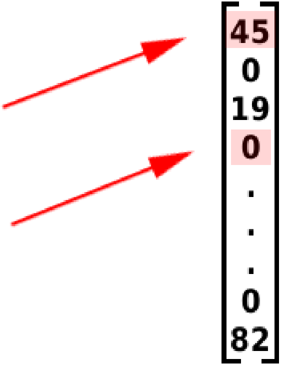
Debugging

Debugging as machine learning

Program Runs as Documents

Predicates	→	Vocabulary
Predicate counts	→	Word counts
Program run	→	Bag-of-words document
Debugging	→	Latent topic analysis

```
int x = my_func()
if (x > 5) {
    branch_42_true++
    ...
}
else {
    branch_42_false++
    ...
}
```



Program Runs as Documents

Predicates	→	Vocabulary
Predicate counts	→	Word counts
Program run	→	Bag-of-words document
Debugging	→	Latent topic analysis

```
int x = my_func()
if (x > 5) {
   branch_42_true++
   ...
}
else {
   branch_42_false++
   ...
}
```

$$\begin{bmatrix} 45 \\ 0 \\ 19 \\ 0 \\ . \\ . \\ . \\ 0 \\ 82 \end{bmatrix}$$

Program Runs as Documents

Predicates	→	Vocabulary
Predicate counts	→	Word counts
Program run	→	Bag-of-words document
Debugging	→	Latent topic analysis

```
int x = my_func()
if (x > 5) {
    branch_42_true++
    ...
}
else {
    branch_42_false++
    ...
}
```

$$\begin{bmatrix} 45 \\ 0 \\ 19 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 82 \end{bmatrix}$$

Program Runs as Documents

Predicates	$\rightarrow$	Vocabulary
Predicate counts	$\rightarrow$	Word counts
Program run	$\rightarrow$	Bag-of-words document
Debugging	$\rightarrow$	Latent topic analysis

```
int x = my_func()
if (x > 5) {
    branch_42_true++
    ...
}
else {
    branch_42_false++
    ...
}
```

$$\begin{bmatrix} 45 \\ 0 \\ 19 \\ 0 \\ . \\ . \\ . \\ 0 \\ 82 \end{bmatrix}$$

Program Runs as Documents

Predicates	→	Vocabulary
Predicate counts	→	Word counts
Program run	→	Bag-of-words document
Debugging	→	Latent topic analysis

```
int x = my_func()
if (x > 5) {
    branch_42_true++
    ...
}
else {
    branch_42_false++
    ...
}
```

$$\begin{bmatrix} 45 \\ 0 \\ 19 \\ 0 \\ . \\ . \\ . \\ 0 \\ 82 \end{bmatrix}$$

No ordering!

Hidden topics

Word counts (observed)

Weighted latent topics (hidden)

$$\begin{bmatrix} \textcolor{blue}{doc\,1} \end{bmatrix} \sim \textcolor{blue}{\theta_1}\begin{bmatrix} \phi_1 \end{bmatrix} + \textcolor{blue}{\theta_2}\begin{bmatrix} \phi_2 \end{bmatrix} + \textcolor{blue}{\theta_3}\begin{bmatrix} \phi_3 \end{bmatrix} + \textcolor{blue}{\theta_4}\begin{bmatrix} \phi_4 \end{bmatrix}$$

$$\begin{bmatrix} \textcolor{green}{doc\,2} \end{bmatrix} \sim \textcolor{green}{\theta_1}\begin{bmatrix} \phi_1 \end{bmatrix} + \textcolor{green}{\theta_2}\begin{bmatrix} \phi_2 \end{bmatrix} + \textcolor{green}{\theta_3}\begin{bmatrix} \phi_3 \end{bmatrix} + \textcolor{green}{\theta_4}\begin{bmatrix} \phi_4 \end{bmatrix}$$

$$\begin{bmatrix} \textcolor{purple}{doc\,3} \end{bmatrix} \sim \textcolor{purple}{\theta_1}\begin{bmatrix} \phi_1 \end{bmatrix} + \textcolor{purple}{\theta_2}\begin{bmatrix} \phi_2 \end{bmatrix} + \textcolor{purple}{\theta_3}\begin{bmatrix} \phi_3 \end{bmatrix} + \textcolor{purple}{\theta_4}\begin{bmatrix} \phi_4 \end{bmatrix}$$

Hidden topics

Word counts (observed)

Weighted latent topics (hidden)

ϕ_3, ϕ_4 = bug topics

$$\begin{bmatrix} \quad doc\,1 \quad \\ \\ \end{bmatrix} \sim \theta_1 \begin{bmatrix} \ \phi_1 \ \\ \\ \end{bmatrix} + \theta_2 \begin{bmatrix} \ \phi_2 \ \\ \\ \end{bmatrix} + \theta_3 \begin{bmatrix} \ \phi_3 \ \\ \\ \end{bmatrix} + \theta_4 \begin{bmatrix} \ \phi_4 \ \\ \\ \end{bmatrix}$$

$$\begin{bmatrix} \quad doc\,2 \quad \\ \\ \end{bmatrix} \sim \theta_1 \begin{bmatrix} \ \phi_1 \ \\ \\ \end{bmatrix} + \theta_2 \begin{bmatrix} \ \phi_2 \ \\ \\ \end{bmatrix} + \theta_3 \begin{bmatrix} \ \phi_3 \ \\ \\ \end{bmatrix} + \theta_4 \begin{bmatrix} \ \phi_4 \ \\ \\ \end{bmatrix}$$

$$\begin{bmatrix} \quad doc\,3 \quad \\ \\ \end{bmatrix} \sim \theta_1 \begin{bmatrix} \ \phi_1 \ \\ \\ \end{bmatrix} + \theta_2 \begin{bmatrix} \ \phi_2 \ \\ \\ \end{bmatrix} + \theta_3 \begin{bmatrix} \ \phi_3 \ \\ \\ \end{bmatrix} + \theta_4 \begin{bmatrix} \ \phi_4 \ \\ \\ \end{bmatrix}$$

Uncover Hidden Bugs

- Goals
  - Uncover hidden bugs
  - **NOT** predicting whether a single run will fail or not
    - ★ It will do that all on its own...
- Assumptions
  - Few hidden bugs $\rightarrow$ many failing runs
  - $\geq$ 1 bug per failing run

Uncover Hidden Bugs

- Goals
  - Uncover hidden bugs
  - **NOT** predicting whether a single run will fail or not
    - It will do that all on its own...
- Assumptions
  - Few hidden bugs $\rightarrow$ many failing runs
  - $\geq$ 1 bug per failing run

Latent Topic Modeling

Debugging	Topic modeling
Predicates	Words
Program runs	Documents
Bug patterns	Topics
Active bug patterns	Topic weights

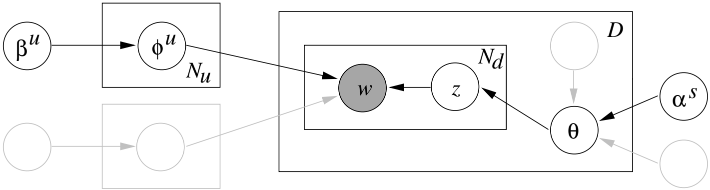
Latent Topic Modeling

Debugging	Topic modeling
Predicates	Words
Program runs	Documents
Bug patterns	Topics
Active bug patterns	Topic weights

Latent Topic Modeling

Debugging	Topic modeling
Predicates	Words
Program runs	Documents
Bug patterns	Topics
Active bug patterns	Topic weights

Latent Dirichlet Allocation (LDA)



For each topic t

$\phi_t \sim \mathrm{Dir}(\beta)$

For each doc d

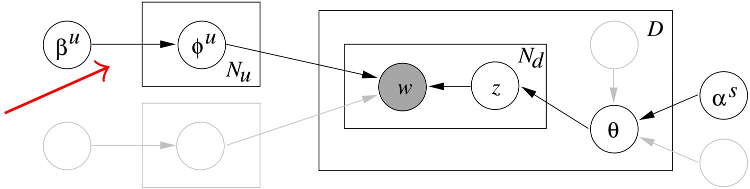
$\theta_d \sim \mathrm{Dir}(\alpha)$

For each word w

Topic $z \sim \theta_d$

$w \sim \phi_z$

Latent Dirichlet Allocation (LDA)



For each topic t

$\phi_t \sim \text{Dir}(\beta)$

For each doc d

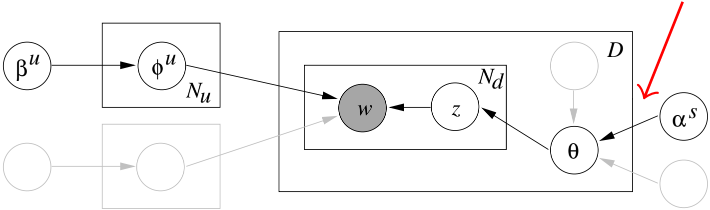
$\theta_d \sim \text{Dir}(\alpha)$

For each word w

Topic $z \sim \theta_d$

$w \sim \phi_z$

Latent Dirichlet Allocation (LDA)



For each topic t

$\phi_t \sim \text{Dir}(\beta)$

For each doc d

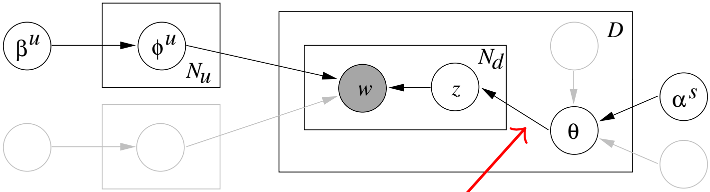
$\theta_d \sim \text{Dir}(\alpha)$

For each word w

Topic $z \sim \theta_d$

$w \sim \phi_z$

Latent Dirichlet Allocation (LDA)



For each topic t

$\phi_t \sim \mathrm{Dir}(\beta)$

For each doc d

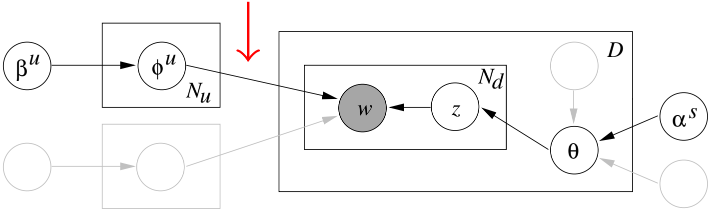
$\theta_d \sim \mathrm{Dir}(\alpha)$

For each word w

Topic $z \sim \theta_d$

$w \sim \phi_z$

Latent Dirichlet Allocation (LDA)



For each topic t

$\phi_t \sim \text{Dir}(\beta)$

For each doc d

$\theta_d \sim \text{Dir}(\alpha)$

For each word w

Topic $z \sim \theta_d$

$w \sim \phi_z$

ΔLDA motivation

LDA cannot recover bug patterns

- Strong non-bug patterns also present
- Most runs successful
  - Usage patterns explain behavior
- Some runs fail
  - Usage patterns mostly explain behavior
  - Usage patterns overwhelm bug patterns

ΔLDA motivation

LDA cannot recover bug patterns

- Strong non-bug patterns also present
- Most runs successful
  - Usage patterns explain behavior
- Some runs fail
  - Usage patterns mostly explain behavior
  - Usage patterns overwhelm bug patterns

ΔLDA motivation

LDA cannot recover bug patterns

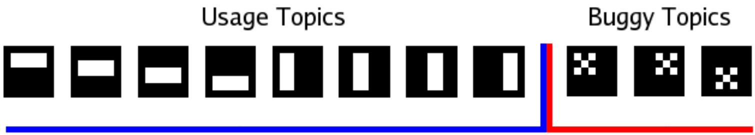
- Strong non-bug patterns also present
- Most runs successful
  - Usage patterns explain behavior
- Some runs fail
  - Usage patterns mostly explain behavior
  - Usage patterns overwhelm bug patterns

ΔLDA motivation

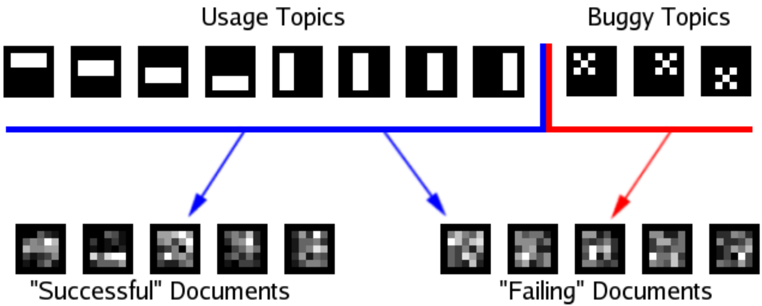
LDA cannot recover bug patterns

- Strong non-bug patterns also present
- Most runs successful
  - Usage patterns explain behavior
- Some runs fail
  - Usage patterns mostly explain behavior
  - Usage patterns overwhelm bug patterns

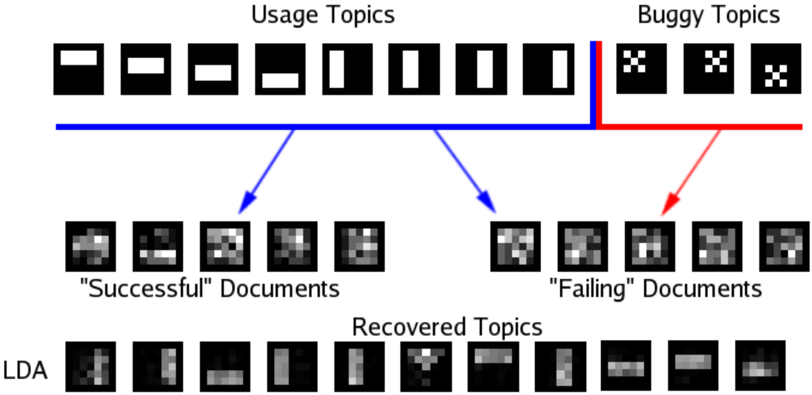
Toy Dataset



Toy Dataset

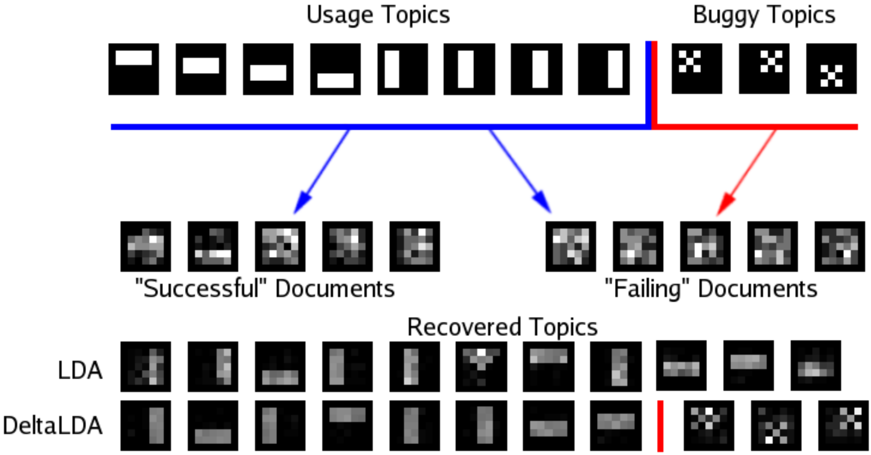

Usage Topics
Buggy Topics
"Successful" Documents
"Failing" Documents

Toy Dataset



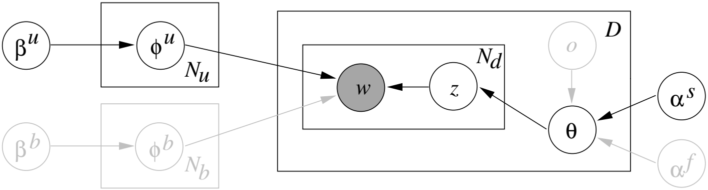
Usage Topics
Buggy Topics
"Successful" Documents
"Failing" Documents
Recovered Topics
LDA

Toy Dataset



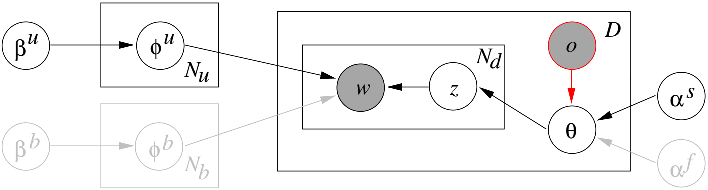
Usage Topics
Buggy Topics
"Successful" Documents
"Failing" Documents
Recovered Topics
LDA
DeltaLDA

ΔLDA



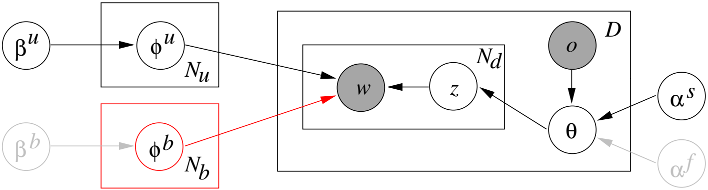
w	words
z	topics
θ	topic weights
ϕ^u	usage topic multinomials
β^u, α^s	hyperparameters
o	outcome flag
ϕ^b	bug topic multinomials
β^b, α^f	hyperparameters

ΔLDA



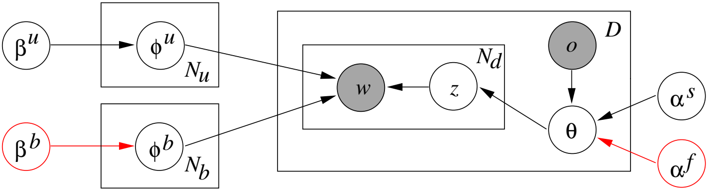
w	words
z	topics
θ	topic weights
ϕ^u	usage topic multinomials
β^u, α^s	hyperparameters
o	outcome flag
ϕ^b	bug topic multinomials
β^b, α^f	hyperparameters

ΔLDA



w	words
z	topics
θ	topic weights
ϕ^u	usage topic multinomials
β^u, α^s	hyperparameters
o	outcome flag
ϕ^b	bug topic multinomials
β^b, α^f	hyperparameters

ΔLDA



w	words
z	topics
θ	topic weights
ϕ^u	usage topic multinomials
β^u, α^s	hyperparameters
o	outcome flag
ϕ^b	bug topic multinomials
β^b, α^f	hyperparameters

ΔLDA

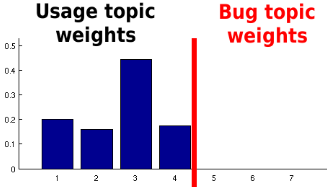
$$\begin{array}{rcl} \text{Successful run } \alpha_s & = & [1\ 1\ 1 \mid 0\ 0] \\ \text{Failing run } \alpha_f & = & [1\ 1\ 1 \mid 1\ 1] \end{array}$$

ΔLDA

$$\text{Successful run } \alpha_s = [1\ 1\ 1\ |\ 0\ 0]$$
$$\text{Failing run } \alpha_f = [1\ 1\ 1\ |\ 1\ 1]$$

Successful run

$p(\theta|\alpha_s) \longrightarrow$

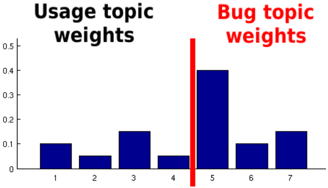


ΔLDA

$$\text{Successful run } \alpha_s = [1\ 1\ 1 \mid 0\ 0]$$
$$\text{Failing run } \alpha_f = [1\ 1\ 1 \mid 1\ 1]$$

Failing run

$p(\theta|\alpha_f) \quad \rightarrow$



Inference

- Need to calculate posterior $p(\mathbf{z}|\mathbf{w}, \mathbf{o})$
- Use $\mathbf{z}$ to estimate
  - Topic multinomials ϕ for each topic
  - Topic weights θ for each document

Collapsed Gibbs Sampling

- Uses easily obtainable counts
- Efficient

$$p(z_k = i|\mathbf{z}_{-k}, \mathbf{w}, \mathbf{o}) \propto \left(\frac{n^i_{-k,j_k} + \beta^i_{j_k}}{n^i_{-k,*} + \sum_{j'}^{W} \beta^i_{j'}} \right) \left(\frac{n^{d_k}_{-k,i} + \alpha^{o_k}_i}{n^{d_k}_{-k,*} + \sum_{i'}^{N_u+N_b} \alpha^{o_k}_{i'}} \right)$$

Inference

- Need to calculate posterior $p(\mathbf{z}|\mathbf{w}, \mathbf{o})$
- Use $\mathbf{z}$ to estimate
  - Topic multinomials ϕ for each topic
  - Topic weights θ for each document

Collapsed Gibbs Sampling

- Uses easily obtainable counts
- Efficient

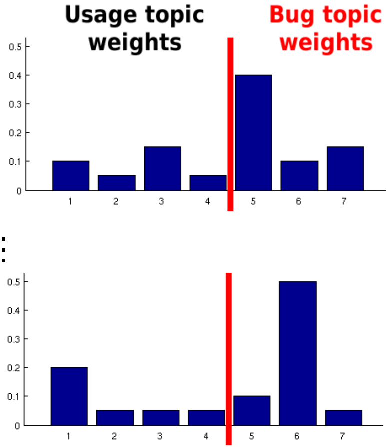
$$p(z_k = i|\mathbf{z}_{-k}, \mathbf{w}, \mathbf{o}) \propto \left(\frac{n^i_{-k,j_k} + \beta^i_{j_k}}{n^i_{-k,*} + \sum_{j'}^{W} \beta^i_{j'}} \right) \left(\frac{n^{d_k}_{-k,i} + \alpha^{o_k}_i}{n^{d_k}_{-k,*} + \sum_{i'}^{N_u+N_b} \alpha^{o_k}_{i'}} \right)$$

Group failing runs by bug

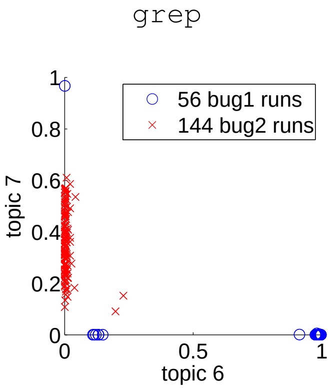
Run 1 θ

$\vdots$ $\vdots$

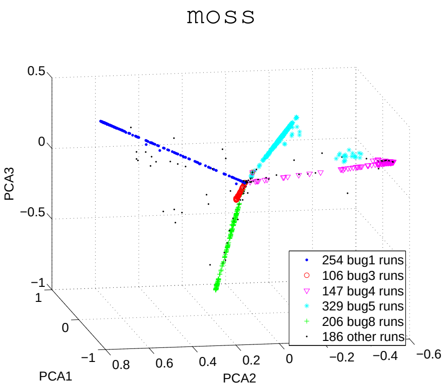
Run n θ



Results



grep
56 bug1 runs
144 bug2 runs
topic 7
topic 6
0
0.2
0.4
0.6
0.8
1
0.5



moss
254 bug1 runs
106 bug3 runs
147 bug4 runs
329 bug5 runs
206 bug8 runs
186 other runs
PCA1
PCA2
PCA3
0.5
0
−0.5
−1
0.8
0.6
0.4
0.2
−0.2
−0.4
−0.6

Results

Rand index wrt ground truth

	ΔLDA	[1]	[2]
`exif`	1.00	0.88	1.00
`grep`	0.97	0.71	0.77
`gzip`	0.89	1.00	1.00
`moss`	0.93	0.93	0.96

Baselines

1. Statistical Debugging: Simultaneous Isolation of Multiple Bugs, Zheng et al ICML '06
2. Scalable Statistical Bug Isolation, Liblit et al PLDI '05

Top predicates for each bug

1. Use ϕ to calculate $p(z|w)$
2. For each bug topic z, sort predicates w by $p(z|w)$

Top predicates for `exif` bug topic 9

Rank	$p(z\|w)$	Predicate w	Expert opinion
1	0.99977	jpeg-data.c:434 jpeg_data_set_exif_data()	Direct result
2	0.69517	jpeg-data.c:436 jpeg_data_set_exif_data()	Direct result
3	0.56748	jpeg-data.c:207 jpeg_data_load_data()	Smoking gun

Top predicates for each bug

1. Use ϕ to calculate $p(z|w)$
2. For each bug topic z, sort predicates w by $p(z|w)$

Top predicates for `exif` bug topic 9

Rank	$p(z\|w)$	Predicate w	Expert opinion
1	0.99977	jpeg-data.c:434 jpeg_data_set_exif_data()	Direct result
2	0.69517	jpeg-data.c:436 jpeg_data_set_exif_data()	Direct result
3	0.56748	jpeg-data.c:207 jpeg_data_load_data()	Smoking gun

Conclusion

- Debugging can be a machine learning problem
- ΔLDA overcomes problems with standard LDA
- ΔLDA successful on real-world data
- Other possible uses of ΔLDA (eg, sentiment analysis)

Acknowledgements
NLM Training Grant 5T15LM07359, NSF Grant CCF-0621487, AFOSR Grant FA9550-07-1-0210

ΔLDA → Debugging

ΔLDA	Debugging
Words	Predicates
Documents	Program runs
Usage topics	Usage patterns
Bug topics	Bug patterns
Bug topic documents	Failing runs
Topic weights	Active usage/bug patterns

Debugging Dataset

Program	Lines of Code	Bugs	Runs	
			Successful	Failing
exif	10,611	2	352	30
grep	15,721	2	609	200
gzip	8,960	2	29	186
moss	35,223	8	1727	1228

Program	Word Types	Topics	
		Usage	Bug
exif	20	7	2
grep	2,071	5	2
gzip	3,929	5	2
moss	1,982	14	8

Joint distribution

$$
\begin{aligned}
p(\mathbf{w}, \mathbf{z}) &= p(\mathbf{w}|\mathbf{z})p(\mathbf{z}) \\
p(\mathbf{w}|\mathbf{z}) &= \\
p(\mathbf{z}) &=
\end{aligned}
$$

Joint distribution

$$
\begin{aligned}
p(\mathbf{w}, \mathbf{z}) &= p(\mathbf{w}|\mathbf{z})p(\mathbf{z}) \\
p(\mathbf{w}|\mathbf{z}) &= \prod_{i}^{N} \int p(\phi_i|\beta) \qquad\qquad \mathrm{d}\phi_i \\
p(\mathbf{z}) &=
\end{aligned}
$$

Joint distribution

$$
\begin{aligned}
p(\mathbf{w}, \mathbf{z}) &= p(\mathbf{w}|\mathbf{z})p(\mathbf{z}) \\
p(\mathbf{w}|\mathbf{z}) &= \prod_{i}^{N} \int p(\phi_i|\beta) \qquad\qquad d\phi_i \\
p(\mathbf{z}) &= \prod_{d}^{D} \int p(\theta_d|\alpha) \qquad\qquad d\theta_d
\end{aligned}
$$

Joint distribution

$$
\begin{aligned}
p(\mathbf{w}, \mathbf{z}) &= p(\mathbf{w}|\mathbf{z})p(\mathbf{z}) \\
p(\mathbf{w}|\mathbf{z}) &= \prod_{i}^{N} \int p(\phi_i|\beta) \qquad\qquad \mathrm{d}\phi_i \\
p(\mathbf{z}) &= \prod_{d}^{D} \int p(\theta_d|\alpha) \prod_{i}^{N} {\theta_{di}}^{n_i^d} \, \mathrm{d}\theta_d
\end{aligned}
$$

Joint distribution

$$
\begin{aligned}
p(\mathbf{w}, \mathbf{z}) &= p(\mathbf{w}|\mathbf{z})p(\mathbf{z}) \\
p(\mathbf{w}|\mathbf{z}) &= \prod_{i}^{N} \int p(\phi_i|\beta) \prod_{j}^{W} {\phi_{ij}}^{n_j^i} \, \mathrm{d}\phi_i \\
p(\mathbf{z}) &= \prod_{d}^{D} \int p(\theta_d|\alpha) \prod_{i}^{N} {\theta_{di}}^{n_i^d} \, \mathrm{d}\theta_d
\end{aligned}
$$