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

The problem seems to occur with NTDETECT.COM.

If this is the first time you have seen this 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: 0x0000000D (0x9E0394C2, 0x08000000, 0x0FBE7E17, 0x08000000)
*** SPCRDC00000 - Address FBFE7E17 base at FBFE0000, datestamp 3066987c
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

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

45
19
82
Program Runs as Documents

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

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

(45 0 19 0 82)
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++
    ...
}
```

```
45
19
0
82
```
Program Runs as Documents

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

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

```
45  0
19  0
  0
  0
  82
```
Program Runs as Documents

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

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

No ordering!

```
45 0 19 0 0 82
```
Hidden topics

Word counts (observed)

\[
\begin{align*}
\text{doc 1} & \sim \theta_1 \phi_1 + \theta_2 \phi_2 + \theta_3 \phi_3 + \theta_4 \phi_4 \\
\text{doc 2} & \sim \theta_1 \phi_1 + \theta_2 \phi_2 + \theta_3 \phi_3 + \theta_4 \phi_4 \\
\text{doc 3} & \sim \theta_1 \phi_1 + \theta_2 \phi_2 + \theta_3 \phi_3 + \theta_4 \phi_4
\end{align*}
\]

Weighted latent topics (hidden)
Hidden topics

Word counts
(observed)

\[
\begin{align*}
\text{doc 1} & \sim \theta_1 \begin{bmatrix} \phi_1 \end{bmatrix} + \theta_2 \begin{bmatrix} \phi_2 \end{bmatrix} + \theta_3 \begin{bmatrix} \phi_3 \end{bmatrix} + \theta_4 \begin{bmatrix} \phi_4 \end{bmatrix} \\
\text{doc 2} & \sim \theta_1 \begin{bmatrix} \phi_1 \end{bmatrix} + \theta_2 \begin{bmatrix} \phi_2 \end{bmatrix} + \theta_3 \begin{bmatrix} \phi_3 \end{bmatrix} + \theta_4 \begin{bmatrix} \phi_4 \end{bmatrix} \\
\text{doc 3} & \sim \theta_1 \begin{bmatrix} \phi_1 \end{bmatrix} + \theta_2 \begin{bmatrix} \phi_2 \end{bmatrix} + \theta_3 \begin{bmatrix} \phi_3 \end{bmatrix} + \theta_4 \begin{bmatrix} \phi_4 \end{bmatrix}
\end{align*}
\]

Weighted latent topics
(hidden)

\[\phi_3, \phi_4 = \text{bug topics}\]
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 → 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 → many failing runs
  - ≥ 1 bug per failing run
## Latent Topic Modeling

<table>
<thead>
<tr>
<th>Debugging</th>
<th>Topic modeling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicates</td>
<td>Words</td>
</tr>
<tr>
<td>Program runs</td>
<td>Documents</td>
</tr>
<tr>
<td>Bug patterns</td>
<td>Topics</td>
</tr>
<tr>
<td>Active bug patterns</td>
<td>Topic weights</td>
</tr>
</tbody>
</table>
Latent Topic Modeling

<table>
<thead>
<tr>
<th>Debugging</th>
<th>Topic modeling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicates</td>
<td>Words</td>
</tr>
<tr>
<td>Program runs</td>
<td>Documents</td>
</tr>
<tr>
<td>Bug patterns</td>
<td>Topics</td>
</tr>
<tr>
<td>Active bug patterns</td>
<td>Topic weights</td>
</tr>
</tbody>
</table>
## Latent Topic Modeling

<table>
<thead>
<tr>
<th>Debugging</th>
<th>Topic modeling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicates</td>
<td>Words</td>
</tr>
<tr>
<td>Program runs</td>
<td>Documents</td>
</tr>
<tr>
<td>Bug patterns</td>
<td>Topics</td>
</tr>
<tr>
<td>Active bug patterns</td>
<td>Topic weights</td>
</tr>
</tbody>
</table>
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 $t$

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

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 $w$
\[ \text{Topic } z \sim \theta_d \]
\[ w \sim \phi_z \]
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 $z \sim \theta_d$

$w \sim \phi_z$
LDA 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
LDA 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
LDA 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
LDA 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

(UW–Madison)
Toy Dataset

Usage Topics

Buggy Topics

"Successful" Documents

"Failing" Documents

Recovered Topics

LDA

(UW–Madison)

Topic-based statistical debugging

ECML 2007 10 / 21
Toy Dataset

Usage Topics

Buggy Topics

"Successful" Documents

"Failing" Documents

Recovered Topics

LDA

DeltaLDA

(UW–Madison)

Topic-based statistical debugging

ECML 2007
w | words
z | topics
θ | topic weights
φ^u | usage topic multinomials
β^u, α^s | hyperparameters
φ^b | bug topic multinomials
β^b, α^f | hyperparameters
\( \Delta LDA \)

\[
\begin{align*}
\beta^u &\rightarrow \phi^u & N_u \\
\beta^b &\rightarrow \phi^b & N_b \\
w &\rightarrow z & N_d \\
\theta &\rightarrow o & \alpha^s \\
\end{align*}
\]

- **w**: words
- **z**: topics
- **\theta**: topic weights
- **\phi^u**: usage topic multinomials
- **\beta^u, \alpha^s**: hyperparameters
- **\phi^b**: bug topic multinomials
- **\beta^b, \alpha^f**: hyperparameters

---

(UW–Madison)  Topic-based statistical debugging  ECML 2007  11 / 21
ALDA

\[
\begin{align*}
\beta^u & \quad \phi^u & \quad N_u \\
\beta^b & \quad \phi^b & \quad N_b
\end{align*}
\]

\[
\begin{align*}
w & \quad z & \quad \theta & \quad \phi^u & \quad \beta^u, \alpha^s & \quad \text{words} \\
& & \quad & \text{topics} & \text{usage topic multinomials} & \text{hyperparameters} \\
& & \quad & \text{topic weights} & \text{hyperparameters} & \text{outcome flag} \\
& & \quad & \phi^b & \text{bug topic multinomials} & \text{hyperparameters}
\end{align*}
\]
**ΔLDA**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w$</td>
<td>words</td>
</tr>
<tr>
<td>$z$</td>
<td>topics</td>
</tr>
<tr>
<td>$\theta$</td>
<td>topic weights</td>
</tr>
<tr>
<td>$\phi^u$</td>
<td>usage topic multinomials</td>
</tr>
<tr>
<td>$\beta^u, \alpha^s$</td>
<td>hyperparameters</td>
</tr>
<tr>
<td>$o$</td>
<td>outcome flag</td>
</tr>
<tr>
<td>$\phi^b$</td>
<td>bug topic multinomials</td>
</tr>
<tr>
<td>$\beta^b, \alpha^f$</td>
<td>hyperparameters</td>
</tr>
</tbody>
</table>

(ΔLDA) Topic-based statistical debugging
Successful run $\alpha_s = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \end{bmatrix}$

Failing run $\alpha_f = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}$
ΔLDA

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

Successful run

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

Usage topic weights

Bug topic weights

(UW–Madison)

Topic-based statistical debugging
Successful run $\alpha_s = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 \end{bmatrix}$

Failing run $\alpha_f = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 \end{bmatrix}$
Inference

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

Collapsed Gibbs Sampling

- Uses easily obtainable counts
- Efficient

$$p(z_k = i|\mathbf{z}_{-k}, \mathbf{w}, \mathbf{o}) \propto \left( \frac{n^i_{-k,jk} + \beta^i_{jk}}{n^i_{-k,*} + \sum_j w_j \beta^i_{jk}} \right) \left( \frac{n^d_{-k,i} + \alpha^o_{ik}}{n^d_{-k,*} + \sum_i N_u + N_b \alpha^o_{ik}} \right)$$
Inference

- Need to calculate posterior $p(z|w, o)$
- Use $z$ to estimate
  - Topic multinomials $\phi$ for each topic
  - Topic weights $\theta$ for each document

Collapsed Gibbs Sampling

- Uses easily obtainable counts
- Efficient

$$p(z_k = i|z_{-k}, w, o) \propto \left( \frac{n^i_{-k,j_k} + \beta^i_{jk}}{n^i_{-k,*} + \sum_j^W \beta^i_j} \right) \left( \frac{n^d_{-k,i} + \alpha^{o_k}_i}{n^d_{-k,*} + \sum_i^{N_u+N_b} \alpha^{o_k}_i} \right)$$
Group failing runs by bug

Run 1

Run n

Usage topic weights

Bug topic weights
Results

```
<table>
<thead>
<tr>
<th>Topic</th>
<th>Runs</th>
</tr>
</thead>
<tbody>
<tr>
<td>topic 6</td>
<td>147 bug4 runs</td>
</tr>
<tr>
<td>topic 7</td>
<td>329 bug5 runs</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Tool</th>
<th>Bug Runs</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>56 bug1 runs</td>
</tr>
<tr>
<td>moss</td>
<td>206 bug8 runs</td>
</tr>
</tbody>
</table>
```

(UW–Madison) Topic-based statistical debugging
Results

Rand index wrt ground truth

<table>
<thead>
<tr>
<th></th>
<th>ΔLDA</th>
<th>[1]</th>
<th>[2]</th>
</tr>
</thead>
<tbody>
<tr>
<td>exif</td>
<td>1.00</td>
<td>0.88</td>
<td>1.00</td>
</tr>
<tr>
<td>grep</td>
<td>0.97</td>
<td>0.71</td>
<td>0.77</td>
</tr>
<tr>
<td>gzip</td>
<td>0.89</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>moss</td>
<td>0.93</td>
<td>0.93</td>
<td>0.96</td>
</tr>
</tbody>
</table>

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 $\phi$ to calculate $p(z|w)$
2. For each bug topic $z$, sort predicates $w$ by $p(z|w)$

| 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 $\phi$ 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
- $\Delta$LDA overcomes problems with standard LDA
- $\Delta$LDA successful on real-world data
- Other possible uses of $\Delta$LDA (eg, sentiment analysis)

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

<table>
<thead>
<tr>
<th>ΔLDA</th>
<th>Debugging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Words</td>
<td>Predicates</td>
</tr>
<tr>
<td>Documents</td>
<td>Program runs</td>
</tr>
<tr>
<td>Usage topics</td>
<td>Usage patterns</td>
</tr>
<tr>
<td>Bug topics</td>
<td>Bug patterns</td>
</tr>
<tr>
<td>Bug topic documents</td>
<td>Failing runs</td>
</tr>
<tr>
<td>Topic weights</td>
<td>Active usage/bug patterns</td>
</tr>
</tbody>
</table>
## Debugging Dataset

<table>
<thead>
<tr>
<th>Program</th>
<th>Lines of Code</th>
<th>Bugs</th>
<th>Runs</th>
<th>Successful</th>
<th>Failing</th>
</tr>
</thead>
<tbody>
<tr>
<td>exif</td>
<td>10,611</td>
<td>2</td>
<td></td>
<td>352</td>
<td>30</td>
</tr>
<tr>
<td>grep</td>
<td>15,721</td>
<td>2</td>
<td></td>
<td>609</td>
<td>200</td>
</tr>
<tr>
<td>gzip</td>
<td>8,960</td>
<td>2</td>
<td></td>
<td>29</td>
<td>186</td>
</tr>
<tr>
<td>moss</td>
<td>35,223</td>
<td>8</td>
<td></td>
<td>1727</td>
<td>1228</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Program</th>
<th>Word Types</th>
<th>Usage</th>
<th>Bug</th>
</tr>
</thead>
<tbody>
<tr>
<td>exif</td>
<td>20</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>grep</td>
<td>2,071</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>gzip</td>
<td>3,929</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>moss</td>
<td>1,982</td>
<td>14</td>
<td>8</td>
</tr>
</tbody>
</table>
Joint distribution

\[ p(w, z) = p(w|z)p(z) \]

\[ p(w|z) = \]

\[ p(z) = \]
Joint distribution

\[
\begin{align*}
p(w, z) &= p(w|z)p(z) \\
p(w|z) &= \prod_i \int p(\phi_i|\beta) \, d\phi_i \\
p(z) &= 
\end{align*}
\]
Joint distribution

\[ p(w, z) = p(w|z)p(z) \]

\[ p(w|z) = \prod_i^N \int p(\phi_i|\beta) \, d\phi_i \]

\[ p(z) = \prod_d^D \int p(\theta_d|\alpha) \, d\theta_d \]
Joint distribution

\[ p(w, z) = p(w|z)p(z) \]

\[ p(w|z) = \prod_{i} \int p(\phi_i|\beta) \, d\phi_i \]

\[ p(z) = \prod_{d} \int p(\theta_d|\alpha) \prod_{i} \theta_i^{n_{id}} \, d\theta_d \]
Joint distribution

\[ p(w, z) = p(w|z)p(z) \]
\[ p(w|z) = \prod_i \int p(\phi_i|\beta) \prod_j \phi_{ij}^{n_i} \, d\phi_i \]
\[ p(z) = \prod_d \int p(\theta_d|\alpha) \prod_i \theta_{di}^{n_i^d} \, d\theta_d \]