Term Filtering with Bounded Error

Zi Yang, Wei Li, Jie Tang, and Juanzi Li

Knowledge Engineering Group
Department of Computer Science and Technology
Tsinghua University, China
{yangzi, tangjie, ljq}@keg.cs.tsinghua.edu.cn
lwthucs@gmail.com

Presented in ICDM’2010, Sydney, Australia
Outline

• Introduction
• Problem Definition
• Lossless Term Filtering
• Lossy Term Filtering
• Experiments
• Conclusion
Introduction

• Term-document correlation matrix
  – $M$: a $|V| \times |D|$ co-occurrence matrix,
  the number of times that term $w_i$ occurs in document $d_j$

– Applied in various text mining tasks
  • text classification [Dasgupta et al., 2007]
  • text clustering [Blei et al., 2003]
  • information retrieval [Wei and Croft, 2006].

– Disadvantage
  • high computation-cost and intensive memory access.
Introduction

- Term-document correlation matrix
  - $M$: a $|V| \times |D|$ co-occurrence matrix, where $|V|$ is the number of terms and $|D|$ is the number of documents.
  - Applied in various text mining tasks:
    - text classification [Dasgupta et al., 2007]
    - text clustering [Blei et al., 2003]
    - information retrieval [Wei and Croft, 2006].

- Disadvantage
  - high computation-cost and intensive memory access.

How can we overcome the disadvantage?
Introduction (cont'd)

• Three general methods:
  – To improve the algorithm itself
  – High performance platforms
    • multicore processors and distributed machines [Chu et al., 2006]
  – Feature selection approaches
    • [Yi, 2003]

We follow this line
• Basic assumption
  – Each term captures more or less information.
• Our goal
  – A subspace of features (terms)
  – Minimal information loss.
• Conventional solution:
  – Step 1: Measure each individual term or the dependency to a group of terms.
    • Information gain or mutual information.
  – Step 2: Remove features with low scores in importance or high scores in redundancy.
Introduction (cont'd)

- Basic assumption
  - Each term captures more or less information.
- Our goal
  - A subspace of features (terms)
    - Minimal information loss.
- Conventional solution:
  - Step 1: Measure each individual term or the dependency to a group of terms.
    - Information gain or mutual information.
  - Step 2: Remove features with low scores in importance or high scores in redundancy.

Limitations
- A generic definition of information loss for a single term and a set of terms in terms of any metric?
- The information loss of each individual document, and a set of documents?

Why should we consider the information loss on documents?
Introduction (cont'd)

• Consider a simple example:

\[ \begin{align*}
\mathbf{w}_A &= (2, 2) \\
\mathbf{w}_B &= (1, 1)
\end{align*} \]

• Question: any loss of information if \( \mathbf{A} \) and \( \mathbf{B} \) are substituted by a single term \( \mathbf{S} \)?
  – Consider only the information loss for terms
    • YES! Consider that \( \epsilon_V \) is defined by some state-of-the-art pseudometrics, cosine similarity.
  – Consider both
    • NO! Because both documents emphasize \( \mathbf{A} \) more than \( \mathbf{B} \). It’s information loss of documents!
Introduction (cont'd)

• Consider a simple example:

  \[
  \mathbf{w}_A = (2, 2)
  \]

What we have done?
- Information loss for both terms and documents
- Term Filtering with Bounded Error problem
- Develop efficient algorithms

• YES! Consider that \( \epsilon_V \) is defined by some state-of-the-art pseudometrics, cosine similarity.

  – Consider both

• NO! Because both documents emphasize \( A \) more than \( B \). It’s information loss of documents!
Outline

- Introduction
- Problem Definition
- Lossless Term Filtering
- Lossy Term Filtering
- Experiments
- Conclusion
Problem Definition - Superterm

• Want
  – less information loss
  – smaller term space

Group terms into superterms!

• Superterm: $s = \{n(s, d_j)\}_j$.

the number of occurrences of superterm $s$ in document $d_j$
Problem Definition – Information loss

- Information loss during term merging?
  - user-specified distance measures $d(\cdot, \cdot)$
    - Euclidean metric and cosine similarity.
  - superterm $s$ substitutes a set of terms $\{w_i\}$
    - information loss of terms $error_V(w) = d(w, s)$.
    - information loss of documents $error_D(d) = d(d, d')$.

A transformed representation for document $d$ after merging

\[
\begin{bmatrix}
\text{Doc 1} & \text{Doc 2} \\
\text{A} & 2 & 2 \\
\text{B} & 1 & 1 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\text{Doc 1} & \text{Doc 2} \\
\text{S} & 1.5 & 1.5 \\
\text{S} & 1.5 & 1.5 \\
\end{bmatrix}
\]

\[
\begin{cases}
error_w = 0 \\
error_D > 0
\end{cases}
\]
Problem Definition – Information loss

• Information loss during term merging?
  – use different distance measures $d(\cdot, \cdot)$
  – superterm $s$ substitutes a set of terms $\{w_i\}$
  • information loss of terms $\text{error}_V(w) = d(w, s)$.
  • information loss of documents $\text{error}_D(d') = d(d, d')$.

Can be chosen with different methods: winner-take-all, average-occurrence, etc.

A transformed representation for document $d$ after merging

$\begin{bmatrix}
A \\
B
\end{bmatrix} \rightarrow \begin{bmatrix}
\text{error}_w = 0 \\
\text{error}_D > 0
\end{bmatrix}$
Problem Definition – TFBE

• Term Filtering with Bounded Error (TFBE)
  – To minimize the size of superterms $|S|$ subject to the following constraints:
    \begin{align*}
    (1) \quad S &= f(V) = \bigcup_{s \in S} s, \\
    (2) \quad \text{for all } w \in V, \quad \text{error}_V(w) &\leq \epsilon_V, \\
    (3) \quad \text{for all } d \in D, \quad \text{error}_D(d) &\leq \epsilon_D.
    \end{align*}
Outline

• Introduction
• Problem Definition
• **Lossless Term Filtering**
• Lossy Term Filtering
• Experiments
• Conclusion
Lossless Term Filtering

• Special case
  – **NO** information loss of any term and document

  \[ \epsilon_V = 0 \text{ and } \epsilon_D = 0 \]

• **Theorem**: The **exact** optimal solution of the problem is yielded by grouping terms of the same vector representation.

• **Algorithm**
  – Step 1: find “local” superterms for each document
    • Same occurrences within the document
  – Step 2: add, split, or remove superterms from global superterm set
Lossless Term Filtering

• Special case
  – **NO** information loss of any term and document

\[ \varepsilon_V = 0 \text{ and } \varepsilon_D = 0 \]

• **Theorem**: The exact optimal solution of the problem is yielded by grouping terms of the

This case is applicable?

• **YES!** On Baidu Baike dataset (containing 1,531,215 documents)
The vocabulary size 1,522,576 -> 714,392 > 50%
Outline

• Introduction
• Problem Definition
• Lossless Term Filtering
• Lossy Term Filtering
• Experiments
• Conclusion
Lossy Term Filtering

- **General case**
  - $\epsilon_V > 0$ and $\epsilon_D > 0$
  - NP-hard!

- **A greedy algorithm**
  - Step 1: Consider the constraint given by $\epsilon_V$.
    - Similar to Greedy Cover Algorithm
  - Step 2: Verify the validity of the candidates with $\epsilon_D$.

- **Computational Complexity**
  - $O(|V|^2 + |V||T||D| + T'(|V|^2 + T|D|))$
  - Parallelize the computation for distances!
Lossy Term Filtering

- General case
  - \( \epsilon_V > 0 \) and \( \epsilon_D > 0 \)
  - NP-hard!

- A greedy algorithm
  - Step 1: Consider the constraint given by \( \epsilon_V \).
  - Similar to Greedy Cover Algorithm
  - Step 2: Verify the validity of the candidates with \( \epsilon_D \).

- Computational Complexity
  - \( O(|V|^2 + |V||T||D| + T'(|V|^2 + T|D|)) \)
  - Parallelize the computation for distances

- Further reduce the size of candidate superterms?
  - Locality-sensitive hashing (LSH) 😊😊
Efficient Candidate Superterm Generation

• LSH

• Basic idea:
  Same hash value (with several randomized hash projections)
  Close to each other (in the original space)

• Hash function for the Euclidean distance
  \[ h_{r,b}(w) = \left\lfloor \frac{r \cdot w + b}{w(\epsilon_V)} \right\rfloor \]
  [Datar et al., 2004].

(Figure modified based on http://cybertron.cg.tu-berlin.de/pdci08/imageflight/nn_search.html)
Efficient Candidate Superterm Generation

• LSH

- Basic idea:
  - Same hash value (with several randomized hash projections)
  - Close to each other (in the original space)

- Hash function for the Euclidean distance
  \[ h(r, w + b) = [r \cdot w + b, w(\epsilon_b)] \]

  - a projection vector
    (drawn from the Gaussian distribution)

  - the width of the buckets (assigned empirically)

(Figure modified based on http://cybertron.cg.tu-berlin.de/pdc08/imag.png)
Efficient Candidate Superterm Generation

• LSH

• Basic idea:

- Same hash value
- Project to hash bucket (randomized)
- Within bucket:
  - Proximity (same width)
- Search neighbors (in bucket)!

Basic idea:

- Hash function for Euclidean distance
  \[ h_r, b_w = r \cdot w + b \]
  \[ w(\epsilon_V) \]

Datar et al., 2004

(Figure modified based on http://cybertron.cg.tu-berlin.de/pdci08/imageflight/nn_search.html)

Now search \( w(\epsilon_V) \)-near neighbors in all the buckets!

the width of the buckets (assigned empirically)
Outline

• Introduction
• Problem Definition
• Lossless Term Filtering
• Lossy Term Filtering
• Experiments
• Conclusion
Experiments

• Settings
  – Datasets
    • Academic: ArnetMiner (10,768 papers and 8,212 terms)
    • 20-Newsgroups (18,774 postings and 61,188 terms)
  – Baselines
    • Task-irrelevant feature selection: document frequency criterion (DF), term strength (TS), sparse principal component analysis (SPCA)
    • Supervised method (only on classification): Chi-statistic (CHIMAX)
Experiments (cont'd)

• Filtering results on Euclidean metric

<table>
<thead>
<tr>
<th>ArnetMiner</th>
<th>$\epsilon_V$</th>
<th>$\epsilon_D$</th>
<th># terms</th>
<th>Term ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>8,212</td>
<td>100%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lossless filter</td>
<td>7,279</td>
<td>88.6%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6,769</td>
<td>82.4%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>4,211</td>
<td>51.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>4,231</td>
<td>51.5%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2,783</td>
<td>33.9%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2,931</td>
<td>35.7%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3,472</td>
<td>42.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>1,694</td>
<td>20.6%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>1,856</td>
<td>22.6%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>2,569</td>
<td>31.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>3,293</td>
<td>40.1%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20-Newsgroups</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Original</td>
<td>61,188</td>
<td>100%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lossless filter</td>
<td>53,406</td>
<td>87.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>47,560</td>
<td>77.7%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>40,791</td>
<td>66.7%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>41,941</td>
<td>68.5%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>35,914</td>
<td>58.7%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>37,451</td>
<td>61.2%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>39,202</td>
<td>64.1%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>33,699</td>
<td>55.1%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>35,089</td>
<td>57.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>35,753</td>
<td>58.4%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>36,479</td>
<td>59.6%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

– Findings

• Errors $\uparrow$, term radio $\downarrow$

• Same bound for terms, bound for documents $\uparrow$, term ratio $\downarrow$

• Same bound for documents, bound for terms $\uparrow$, term ratio $\downarrow$
Experiments (cont'd)

• The information loss in terms of error

- Average term error
- Max term error
- Average document error
- Max document error

Legend:
- DF
- TS
- SPCA
- Greedy LF
Experiments (cont'd)

• Efficiency Performance of Parallel Algorithm and LSH*

* Optimal resulting from different choice of $w(\epsilon_V)$ tuned
Experiments (cont'd)

- Applications
  - Clustering (Arnet)
  - Classification (20NG)
  - Document retrieval (Arnet)
Outline

• Introduction
• Problem Definition
• Lossless Term Filtering
• Lossy Term Filtering
• Experiments
• Conclusion
Conclusion

• Formally define the problem and perform a theoretical investigation
• Develop efficient algorithms for the problems
• Validate the approach through an extensive set of experiments with multiple real-world text mining applications
Thank you!
References