Social Movement Understanding by Keyword Tracking

Virach Sornlertlamvanich
virach@gmail.com
Kobkrit Viriyayudhakorn
kobkrit@gmail.com

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
THAILAND

67,091,000
TOTAL POPULATION

24,000,000
INTERNET USERS

16,834,140
USERS ON TOP SOCIAL NETWORK

78,667,910
MOBILE SUBSCRIBERS

34%
URBAN

66%
RURAL

36%
INTERNET PENETRATION

25%
SOCIAL NETWORKING PENETRATION

117%
MOBILE PENETRATION

http://tulaneict4d.wordpress.com/2013/04/05/social-media-in-thailand/

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
Thailand Monthly Active Users on Social Network and Messenger App in 2013

Data Preparation

- Word segmentation
- Keyword extracted from topic related documents (training set)
- Tweeter inquiry using the prepared topic related list of keywords
- Text similarity using GETA algorithm
GETA: Association Search Engine

- Proposed by Akihiko Takano (Takano, 2003)
- Back-and-forth searches between words and documents spaces
  - Set of keywords $\rightarrow$ The top-n ranks of associated documents
  - Set of documents $\rightarrow$ The top-m ranks of associated keywords
- Require a **Word Article Matrix (WAM)**, and a similarity function

![Diagram of GETA: Association Search Engine](image)

**Keywords** $\rightarrow$ **Top-n Ranked Associated Documents**

**Top-m Ranked Associated Keywords** $\rightarrow$ **Documents space**

![Diagram of GETA: Association Search Engine](image)

**Words space**

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
GETA: Association Search Engine

- Proposed by Akihiko Takano (Takano, 2003)
- Back-and-forth searches between words and documents spaces
  - Set of *keywords* → The top-n ranks of *associated documents*
  - Set of *documents* → The top-m ranks of *associated keywords*
- Require a **Word Article Matrix (WAM)**, and a **similarity function**

![Diagram showing GETA: A Word Article Matrix (WAM) and a similarity function connected to keywords and associated documents.](image)
Word Article Matrix (WAM)

Building WAM is indexing

All Wikipedia Articles

All Pages’ Contents

All Pages’ Titles

Word seg. & Lemmatization
Thinking -> Think

Getting

A Word Article Matrix
(WAM)

A similarity function

Top-Ranked
Associated
Keywords

Documents space

Keywords

Words space

Pages\Words | “Twitter” | “Tennis” | “Dollar” | “Google” | ...
---|---|---|---|---|---
IT | 2 | 0 | 1 | 4 | ...
Sport | 0 | 2 | 1 | 0 | ...
Economics | 0 | 0 | 2 | 0 | ...
...

A Wikipedia WAM
EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
Similarity Functions

- Ranking weight in both words and documents spaces
- Equation form

\[
SIM(d, q) = \sum_{t \in q} \frac{wq(t, q) \cdot wd(t, d)}{\text{norm}(d, q)}
\]

- \(wd(t,d) = \) Weight of term \(t\) in document \(d\), \(wd(t, d) = 0 \text{ if } t \notin d\)
- \(wq(t,q) = \) Weight of term \(t\) in query sentence \(q\), \(wq(t, q) = 0 \text{ if } t \notin q\)
- \(\text{norm}(d,q) = \) Normalization function due to the different length of \(d\) and \(q\).

- Smart measure (Singhal et.al., 1996)

\[
\frac{1}{avg(f_d) + \theta(f_d - avg(f_d))} \sum_{t \in q \land t \in d} \log\left(\frac{N}{f_t}\right) \cdot \frac{1 + \log(f_{d,t})}{1 + \log(avg_{d}(f_{d,t}))} \cdot \frac{1 + \log(f_{q,t})}{1 + \log(avg_{q}(f_{q,t}))}
\]

- Dot Product (Wilkinson et.al., 1996)

\[
\sum_{t \in q \land t \in d} (w_{q,t} \cdot w_{d,t})
\]
Example GETA Calculation

SIM Function: Dot Product

\[ \sum_{t \in \text{related}} (w_{q,t} \cdot w_{d,t}) \]

**A idea label**
“Twitter has 800M dollars”

A Word Article Matrix (WAM)

A similarity function

Top-Ranked Associated Documents

Words space

Documents space

(n=2) Select Top-n Most Associated Documents Ranking

<table>
<thead>
<tr>
<th>Pages</th>
<th>“Twitter”</th>
<th>“Tennis”</th>
<th>“Dollar”</th>
<th>“Google”</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Sport</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Economics</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Wikipedia WAM

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
## Example GETA Calculation

**SIM Function: Dot Product**

\[
\sum_{t \equiv n \in d} (w_{q,t} \cdot w_{d,t})
\]

1. **Word seg. & Lemmatization**
   - **dollar**
   - "Twitter has 800M dollars"
   - A idea label

2. **Wikipedia WAM**

<table>
<thead>
<tr>
<th>Pages/Words</th>
<th>&quot;Twitter&quot;</th>
<th>&quot;Tennis&quot;</th>
<th>&quot;Dollar&quot;</th>
<th>&quot;Google&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Sport</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Economics</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

3. **Dot Product**

<table>
<thead>
<tr>
<th>Pages</th>
<th>Dot Product Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>3</td>
</tr>
<tr>
<td>Sport</td>
<td>0</td>
</tr>
<tr>
<td>Economics</td>
<td>2</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Keywords**
- Top-Ranked Associated Documents
- A Word Article Matrix (WAM)
- A similarity function

**Words space**

**Documents space**

**EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany**
Example GETA Calculation (Cont.)

SIM Function: Dot Product $\sum_{t \in q \land t \in d} (w_{q,t} \cdot w_{d,t})$

Top-2 Most Associated Documents

<table>
<thead>
<tr>
<th>Pages</th>
<th>Dot Product Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>3</td>
</tr>
<tr>
<td>Sport</td>
<td>0</td>
</tr>
<tr>
<td>Economics</td>
<td>2</td>
</tr>
</tbody>
</table>

Output of GD

Wikipedia WAM

(m=3) Top-m Most Associated Keywords

<table>
<thead>
<tr>
<th>“Twitter”</th>
<th>“Tennis”</th>
<th>“Dollar”</th>
<th>“Google”</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0</td>
<td>7</td>
<td>12</td>
</tr>
</tbody>
</table>
Example GETA Calculation (Cont.)

SIM Function: Dot Product $\sum_{t=q \wedge \ell \in d} (w_{q,t} \cdot w_{d,t})$

Top-2 Most Associated Documents

<table>
<thead>
<tr>
<th>Pages</th>
<th>Dot Product Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>3</td>
</tr>
<tr>
<td>Sport</td>
<td>0</td>
</tr>
<tr>
<td>Economics</td>
<td>2</td>
</tr>
</tbody>
</table>

*Sport Pages = 0, since only Top-2 is selected from previous slide.

A idea label “Twitter has 800M dollars”

Google, Dollar, Twitter

Wikipedia WAM

(m=3) Top-m Most Associated Keywords

<table>
<thead>
<tr>
<th>“Twitter”</th>
<th>“Tennis”</th>
<th>“Dollar”</th>
<th>“Google”</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0</td>
<td>7</td>
<td>12</td>
</tr>
</tbody>
</table>

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
Tweeter Viewer

Word Article Matrix (WAM)

Social movement timeline viewer

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
Coup on May 22, 2014

- military, NCPO, country, announce, peace, power, government, coup d’état, gathering, police, situation, PM, control, seize, meeting, economy, law, war, leader, minister, election, democracy, revolution, seize the power, curfew, martial law
Tweet Query

• Search Tweets by using Restful API
  – GET search/tweets. Set q = the keyword set
  – 100 tweets/search limited
  – Repeatedly fetch data until all tweets in the coup periods are discovered

• Be able to search back to 7 days

May 22, 2014 Coup-related tweet : 339,148 tweets
Timeline Word Cloud

Coup D'etat

EJC2014 Panel Session, June 3-6, 2014, Kiel, Germany
Conclusion

• Key word expansion is effective to understand the short message i.e. tweet

• Key word expansion can be done using the known training corpus and GETA algorithm

• Timeline word cloud shows the development of the social movement e.g. some events are predictable