Graph-of-words: boosting text mining with graphs

F. Rousseau, M. Vazirgiannis, A. Tixier

DaSciM team, LIX, École Polytechnique, France

ParisBD2016
March 24, 2016
Text is everywhere. For instance:

✓ search engines
✓ marketing and advertising
✓ social media (tweets, posts, blogs)
✓ virtual meetings (speech to text, chat)
✓ big proprietary databases (injury reports, insurance claims, customer complaints...)

The Machine Learning tasks are numerous:

✓ **summarization** (e.g., keywords, paragraph, topics)
✓ **classification** (e.g., sentiment analysis)
✓ **information retrieval** (answer user queries)
✓ **(sub)event/topic detection from text streams** (e.g., natural disaster, topic discussed...)
✓ **link prediction** (e.g., in citation networks)
Limitations of Bag-of-Words

✓ traditional representation of text (with TF or TF-IDF weighting)
✓ assumes independence between terms
✓ does not capture term order (**Mary is quicker than John** = **John is quicker than Mary**)

**information retrieval is the activity of obtaining information resources relevant to an information need from a collection of information resources**

(activity,1), (collection,1), (information,4), (relevant,1), (resources,2)
A method for solution of systems of linear algebraic equations with m-dimensional lambda matrices. A system of linear algebraic equations with m-dimensional lambda matrices is considered. The proposed method of searching for the solution of this system lies in reducing it to a numerical system of a special kind.
Graph-of-Words

✓ captures term dependence
✓ encodes the strength of the dependence as edge weights
✓ captures term order (via directed edges)
✓ recently reached state-of-the-art on many NLP tasks:

- information retrieval [Rousseau and Vazirgiannis, 2013]
- document classification [Nikolentzos et al. 2016, Rousseau et al., 2015; Malliaros and Skianis, 2015]
- **single-document keyword extraction** [Rousseau and Vazirgiannis, 2015]
Graph degeneracy – concept of k-core

- a **core** of order $k$ (or $k$-core) of a graph $G$ is a maximal connected subgraph of $G$ in which every vertex $v$ has at least degree $k$ [Seidman, 1983]

- the $k$-core **decomposition** of $G$ is the list of all its cores from 0 ($G$ itself) to $k_{\text{max}}$ (its main core) ⇒ hierarchy of levels of increasing cohesiveness

- linear (resp. linearithmic) time algorithm available for unweighted (resp. weighted) edges [Batagelj and Zaveršnik, 2002]

- the **core number** of a node is the highest order of a core that contains this node
Illustration of k-core decomposition

Graph-of-words: boosting text mining with graphs
Main Core Retention on Graph-of-Words for Keyword Extraction

⇒ **simple idea**: represent a document as a **graph-of-words**, degenerate the graph, and then, retain the members of the main core of the graph as the keywords

⇒ this approach extracts keywords based on their centrality but also their **cohesiveness** in the graph-of-words
A method for solution of systems of linear algebraic equations with m-dimensional lambda matrices.

Keywords manually assigned by human annotators
linear algebra equat; numer system; m-dimension lambda matric

WK-core PageRank

<table>
<thead>
<tr>
<th>Keyword</th>
<th>WK-core</th>
<th>PageRank</th>
</tr>
</thead>
<tbody>
<tr>
<td>system</td>
<td>6</td>
<td>system</td>
</tr>
<tr>
<td>matric</td>
<td>6</td>
<td>matric</td>
</tr>
<tr>
<td>lambda</td>
<td>6</td>
<td>solut</td>
</tr>
<tr>
<td>linear</td>
<td>6</td>
<td>lambda</td>
</tr>
<tr>
<td>equat</td>
<td>6</td>
<td>linear</td>
</tr>
<tr>
<td>algebra</td>
<td>6</td>
<td>equat</td>
</tr>
<tr>
<td>m-dimension</td>
<td>6</td>
<td>algebra</td>
</tr>
<tr>
<td>method</td>
<td>5</td>
<td>m-dimension</td>
</tr>
<tr>
<td>solut</td>
<td>5</td>
<td>propos</td>
</tr>
<tr>
<td>propos</td>
<td>4</td>
<td>method</td>
</tr>
<tr>
<td>numer</td>
<td>3</td>
<td>special</td>
</tr>
<tr>
<td>specia</td>
<td>2</td>
<td>numer</td>
</tr>
<tr>
<td>kind</td>
<td>2</td>
<td>kind</td>
</tr>
</tbody>
</table>
Experiments: set-up

2 standard datasets:

- *Hulth2003* – 500 abstracts from the *Inspec* database [Hulth, 2003]
- *Krapi2009* – 2,304 ACM full papers in Computer Science (references and captions excluded) [Krapivin et al., 2009]

Each document has a set of golden keywords assigned by humans
⇒ *precision*, *recall* and *F1-score* per document
⇒ *macro-average* each metric at the collection level

Comparisons:
- PageRank
- HITS (authority scores only) \{ top 33% or top 15 keywords \}
- K-core
- Weighted K-core \{ main core \}
### Experiments: results

<table>
<thead>
<tr>
<th>Graph</th>
<th>Dataset</th>
<th>Macro-average F1-score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PageRank</td>
</tr>
<tr>
<td>undirected edges</td>
<td>Hulth2003</td>
<td>47.32</td>
</tr>
<tr>
<td></td>
<td>Krapi2009</td>
<td>49.59</td>
</tr>
<tr>
<td>forward edges</td>
<td>Hulth2003</td>
<td>45.70</td>
</tr>
<tr>
<td></td>
<td>Krapi2009</td>
<td>45.72</td>
</tr>
<tr>
<td>backward edges</td>
<td>Hulth2003</td>
<td>47.57</td>
</tr>
<tr>
<td></td>
<td>Krapi2009</td>
<td>50.51</td>
</tr>
</tbody>
</table>

**Table:** Macro-average F1-score for PageRank, HITS, K-core and Weighted K-core (WK-core). Bold font marks the best performance in a block of a row. * indicates statistical significance at $p < 0.05$ using the Student's t-test w.r.t. the PageRank baseline of the same block of the same row.
Conclusion

- extracting the main core captures a cohesive subgraph of vertices that are not only central but also densely connected

- leads to better performance, in terms of F1 score but also adaptability (number of keywords adapt to graph size, i.e., document size)
Interactive web demo

https://safetyapp.shinyapps.io/GoWvis/

- graph-of-words interactive visualization
- many text preprocessing, graph building and graph mining tuning parameters
- keyword extraction
- extractive summarization
Thank you for your attention
Questions?
Word clustering using PLSA enhanced with long distance bigrams.

Graph-based term weighting for information retrieval.

LexRank: graph-based lexical centrality as salience in text summarization.

Improved automatic keyword extraction given more linguistic knowledge.

Keeping keywords fresh: A BM25 variation for personalized keyword extraction.

Large dataset for keyphrases extraction.
Technical Report DISI-09-055, University of Trento.

Graph-based keyword extraction for single-document summarization.
References II

Do summaries help?

TextRank: Bringing order into texts.

Automatic summarization.

Graph-of-word and TW-IDF: New approach to ad hoc IR.

Network structure and minimum degree.
Social Networks, 5:269–287.

Learning to extract keyphrases from text.

Fast generation of result snippets in web search.