Graph-of-words: boosting text mining with graphs

F. Rousseau, M. Vazirgiannis, **A. Tixier**

DaSciM team, LIX, École Polytechnique, France

ParisBD2016

March 24, 2016

Context

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

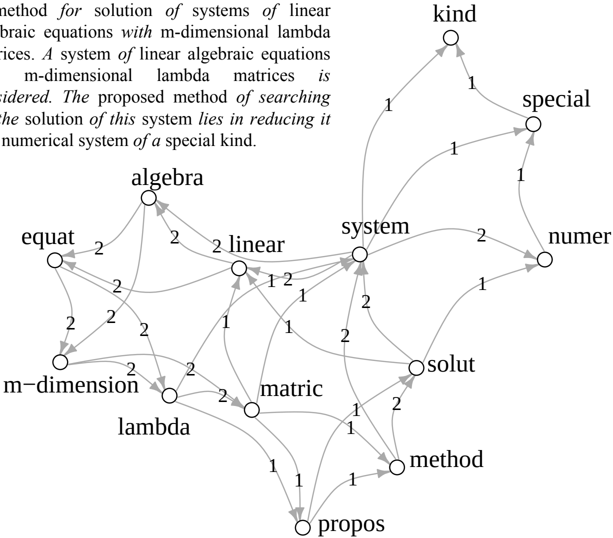
- ✓ traditional representation of text (with TF or TF-IDF weighting)
- ✓ assumes independence between terms
- ✓ does not capture term order (*Mary is quicker than John* = *John is quicker than Mary*)

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)...

Graph-of-Words: a novel approach for text mining

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

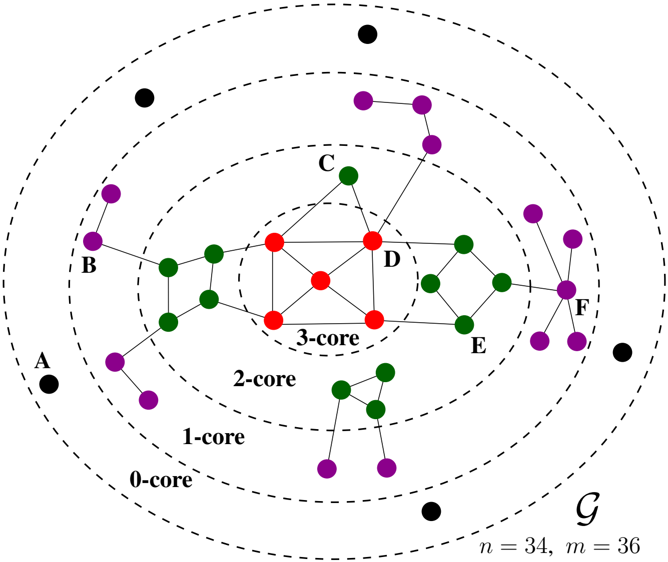
- a **core** of order k (or k-core) of a graph G is a maximal connected subgraph of G in which every vertex v has at least degree k [Seidman, 1983]

- the k-core **decomposition** of G is the list of all its cores from 0 (G itself) to k_{max} (its main core)
$\Rightarrow$ 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

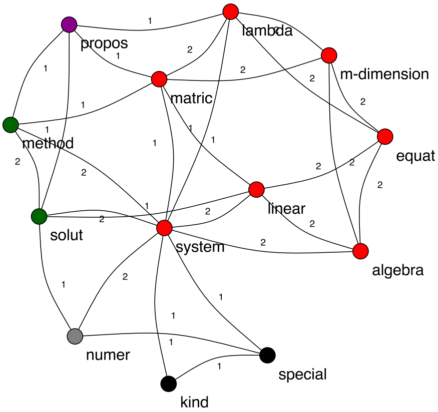


Main Core Retention on Graph-of-Words for Keyword Extraction

$\Rightarrow$ **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

$\Rightarrow$ this approach extracts keywords based on their centrality but also their **cohesiveness** in the graph-of-words

Illustration of main core vs. PageRank



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

WK-core		PageRank	
system	6	**system**	1.93
matric	6	**matric**	1.27
lambda	6	solut	1.10
linear	6	**lambda**	1.08
equat	6	**linear**	1.08
algebra	6	**equat**	0.90
m-dim...	6	**algebra**	0.90
method	5	**m-dim...**	0.90
solut	5	propos	0.89
propos	4	method	0.88
numer	3	special	0.78
specia	2	**numer**	0.74
kind	2	kind	0.55

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

$\Rightarrow$ **precision**, **recall** and **F1-score** per document

$\Rightarrow$ **macro-average** each metric at the collection level

Comparisons:

- PageRank
- HITS (authority scores only)

$\left.\right\}$ top 33% or top 15 keywords

- K-core
- Weighted K-core

$\left.\right\}$ main core

Experiments: results

Graph	Dataset	Macro-average F1-score (%)			
		PageRank	HITS	K-core	WK-core
undirected edges	Hulth2003	47.32	46.62	49.06*	**51.92***
	Krapi2009	49.59	47.96	46.61	**50.77***
forward edges	Hulth2003	45.70	45.03	**51.65***	50.59*
	Krapi2009	45.72	44.95	46.03	**47.01***
backward edges	Hulth2003	47.57	45.37	45.20	**50.03***
	Krapi2009	**50.51**	47.38	46.93	50.42

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?

References I

Bassiou, N. and Kotropoulos, C. (2010).
Word clustering using PLSA enhanced with long distance bigrams.
In *Proceedings of the 20th International Conference on Pattern Recognition*, ICPR '10, pages 4226–4229.

Blanco, R. and Lioma, C. (2012).
Graph-based term weighting for information retrieval.
Information Retrieval, 15(1):54–92.

Erkan, G. and Radev, D. R. (2004).
LexRank: graph-based lexical centrality as salience in text summarization.
Journal of Artificial Intelligence Research, 22(1):457–479.

Hulth, A. (2003).
Improved automatic keyword extraction given more linguistic knowledge.
In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '03, pages 216–223.

Karkali, M., Plachouras, V., Stefanatos, C., and Vazirgiannis, M. (2012).
Keeping keywords fresh: A BM25 variation for personalized keyword extraction.
In *Proceedings of the 2nd Temporal Web Analytics Workshop*, TempWeb '12, pages 17–24.

Krapivin, M., Autaeu, A., and Marchese, M. (2009).
Large dataset for keyphrases extraction.
Technical Report DISI-09-055, University of Trento.

Litvak, M. and Last, M. (2008).
Graph-based keyword extraction for single-document summarization.
In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, MMIES '08, pages 17–24.

References II

McKeown, K., Passonneau, R. J., Elson, D. K., Nenkova, A., and Hirschberg, J. (2005).
Do summaries help?
In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 210–217.

Mihalcea, R. and Tarau, P. (2004).
TextRank: Bringing order into texts.
In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '04, pages 404–411.

Nenkova, A. and McKeown, K. R. (2011).
Automatic summarization.
Foundations and Trends in Information Retrieval, 5(2-3):103–233.

Rousseau, F. and Vazirgiannis, M. (2013).
Graph-of-word and TW-IDF: New approach to ad hoc IR.
In *Proceedings of the 22nd ACM international conference on Information and knowledge management*, CIKM '13, pages 59–68.

Seidman, S. B. (1983).
Network structure and minimum degree.
Social Networks, 5:269–287.

Turney, P. D. (1999).
Learning to extract keyphrases from text.
Technical report, National Research Council of Canada, Institute for Information Technology.

Turpin, A., Tsegay, Y., Hawking, D., and Williams, H. E. (2007).
Fast generation of result snippets in web search.
In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 127–134.