ÉCOLE POLYTECHNIQUE
UNIVERSITÉ PARIS-SACLAY

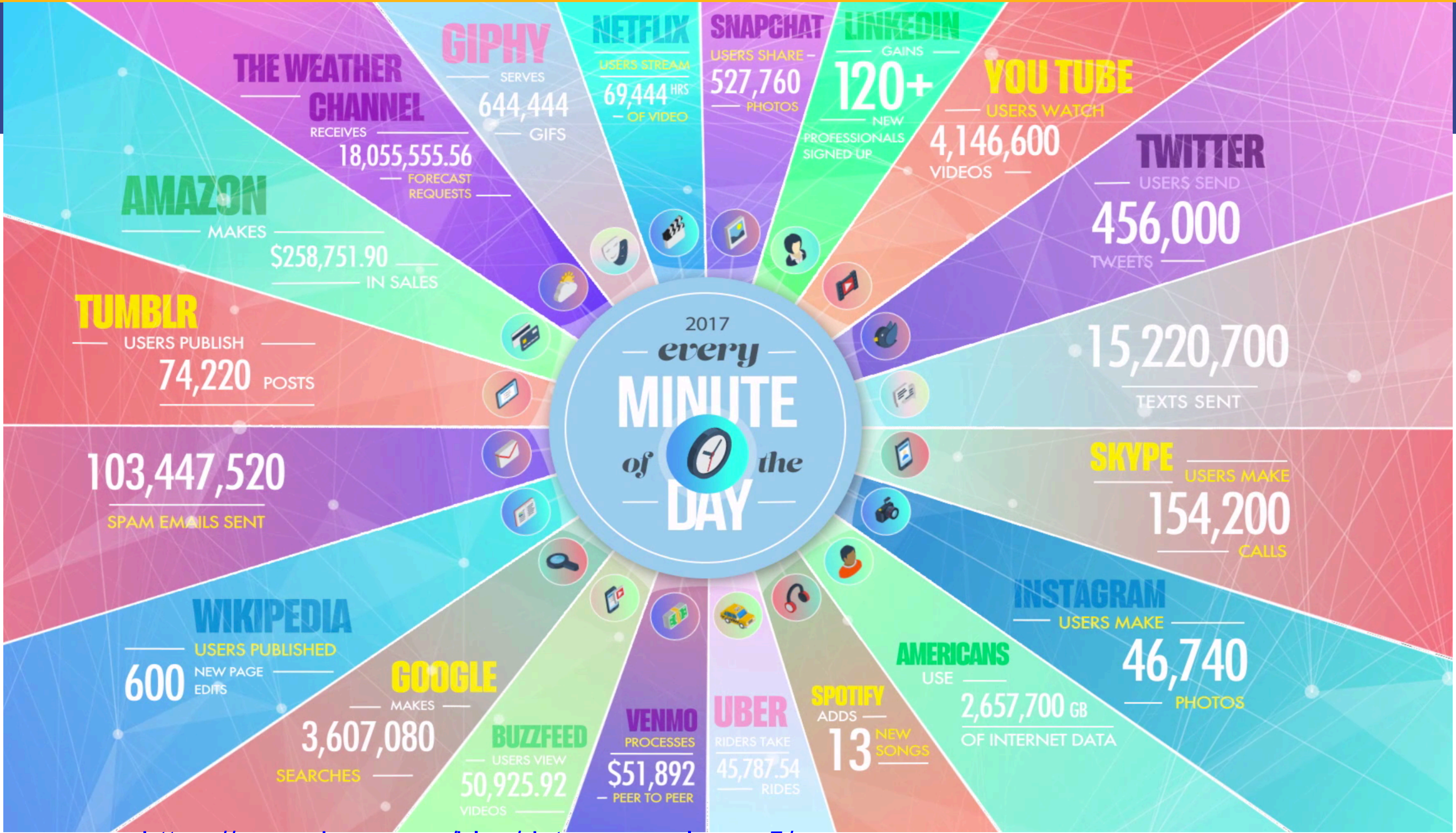
Graph Degeneracy and applications

Michalis Vazirgiannis

Data Science and Mining group, École Polytechnique
http://www.lix.polytechnique.fr/~mvazirg
https://www.lix.polytechnique.fr/dascim
Twitter: @mvazirg

September 2018

Gentle Introduction to Bigdata and Machine Learning



https://www.domo.com/blog/data-never-sleeps-5/

How big are the data?

25 petabytes per year – 25.000 1TB hard disks

100 Petabytes Since 2012 in videos and photos
Daily : **2.7 billion likes**

Estimated **1 Exabyte** from historical data of customers

10 Exabytes from billions of requests

These data are definitely "Big"

- Is this relevant for every business?
- **What is "Big Data"?**

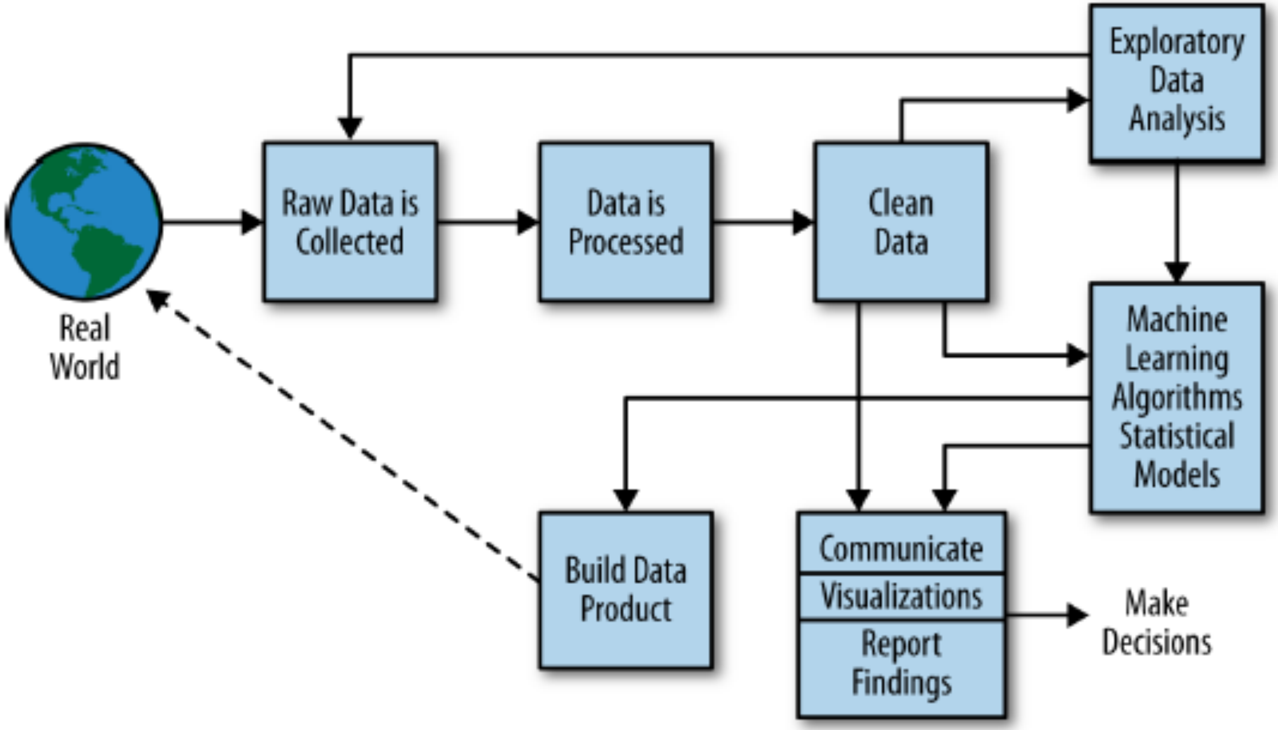
Storage Resources

- Storage capacity has increased massively over the years, BUT **not access speeds**
- 1 disk to read 1TB (100 Mb/s) 25 minutes
  - > Even more to write
- Solution: multiple disks in parallel
- 100 drives -> less than two minutes to read 1TB
  - In one machine? How big?
  - What about protection from hardware failure (20%/4 years)?
  - Data Storage is not trivial!

Distributed File System

- One Machine does not scale
  - Instead **many** working as one
    - If you want more resources add more machines
- **H**adoop **D**istributed **F**ile **S**ystem
  - The most popular technology for Big Data
  - The machines don’t have to be uniform
  - More than just the file system
    - Map-Reduce
  - major Big Data technologies built on top of Hadoop

Data Science Life cycle



Doing Data Science - O'Reilly Media

Competitions | Kaggle - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

Mozilla Firefox Start Page | CIKM 2013, Burlingame, CA, USA | eClass του Οικονομικού Πανεπιστημί... | Competitions | Kaggle | Kaggle Member FAQ

www.kaggle.com/competitions

καγγλε

kaggle

Customer Solutions ▾ | Competitions | Community ▾ | Sign Up | Login

Welcome to Kaggle, the leading platform for predictive modeling competitions. Here's how to jump into competing on Kaggle —

New to Data Science? Visit our Wiki »
Learn about hosting a competition »
in-Class & Research competitions »

Enter

Find a competition & download the training data. You don't need new software/skills to submit.

Build

Build a model using whatever methods you prefer and upload your predictions to Kaggle.

...Win!

Kaggle scores your solution in real time and you'll see your place on the live leaderboard.

Active Competitions

All Competitions

116 found, 14 active

Search competitions

- All competitions
- Enterable

Status

- Active

Competition Name	Reward	Teams	Deadline
 **Titanic: Machine Learning from Disaster** Predict survival on the Titanic (with tutorials in Excel, Python and an introduction to Random Forests)	Knowledge	6876	11 months
 **Digit Recognizer** Classify handwritten digits using the famous MNIST data	Knowledge	1947	9 months
Data Science London **Data Science London + Scikit-learn** Scikit-learn is an open-source machine learning library for Python. Give it a try here!	Knowledge	501	4 months
 **Dogs vs. Cats**			

Find: Vazirg | Next | Previous | Highlight all | Match case

EL 3:50 μμ 9/10/2013

Machine learning

- Tom Mitchell(1998): Well-posed Learning Problem: A computer program is said to learn from experience **E** with respect to some task **T** and some performance measure **P**, if its performance on T, as measured by P, *improves with experience* E.

email spam learning

- Task: email classification to spam/no-spam
- Experience: the user’s action to characterize emails
- Performance: # of emails characterized as spam correctly.

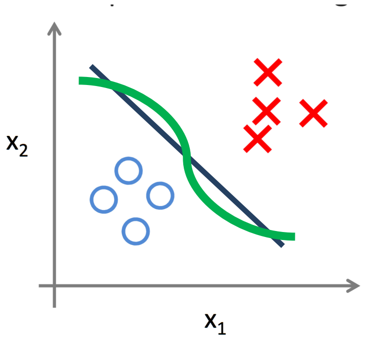
Applications of Machine Learning

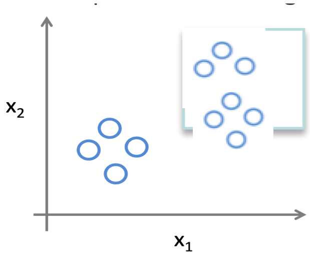
- Text or document classification, e.g., spam detection;
- Natural language processing, e.g., morphological analysis, part-of-speech tagging, statistical parsing, named-entity recognition
- Recommendation systems, search engines, information extraction systems
- Fraud detection (credit card, telephone) and network intrusion
- Speech recognition, speech synthesis, speaker verification;
- Optical character recognition (OCR);
- Computational biology applications, e.g., protein function or structured prediction, Medical diagnosis;
- Computer vision tasks, e.g., image recognition, face detection;
- Games, e.g., chess, backgammon;
- Unassisted vehicle control (robots, navigation);
- …

ML Tasks

Main Tasks

- Supervised Learning - Approximation
- Unsupervised Learning – Description





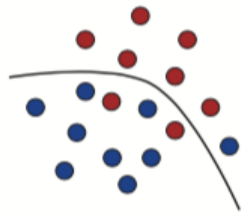
- Supervised & unsupervised learning synergy

DATA → Description → Summaries → Function approximation → LABELS

Unsupervise
d learning

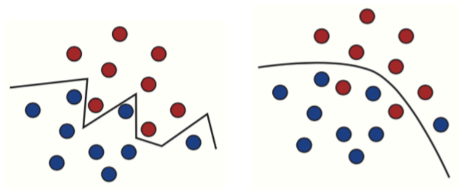
Supervised
learning

Machine Learning example



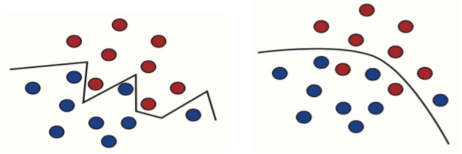
- Red and blue dots - training set
- Red/Blue - labels/classes
- Features: the space in which the training set is embedded (i.e. the (x,y) coordinates for this example)
- Objective: Learn a model (a function) *f* that based on the position of a sample decides the class of the point.
- Test sample: Examples to evaluate the performance of a learning algorithm - separate from the training and not made available in the learning stage

Machine Learning example



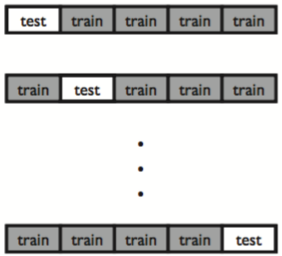
- Loss function: A function L that measures the error, or loss, between a predicted and a true label. If y/y' the true/predicted labels:
  - Square loss: $$E = \sum_{i=1}^{k} (y(i) - y'(i))^2$$
  - Other loss functions: Hinge, Logistic, Cross entropy…
- Hypothesis set: set of functions mapping features to labels (i.e. points to blue/red)
- Over fitting vs generalization: a function may be consistent (i.e. zero training error) but not generalize well.

Machine Learning example



- Cross-validation: in many cases there are not enough training data.
  - Split the m data into n subsets(folds) and let θ the model parameters
  - Train the algorithm for n-1 folds and test on the n-th
  - Compute the cross validation error
  - Choose parameters θ that

  minimize the cv. error



$$\widehat{R}_{\mathrm{CV}}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{\frac{1}{m_i}\sum_{j=1}^{m_i} L(h_i(x_{ij}), y_{ij})}_{\text{error of } h_i \text{ on the } i\text{th fold}} .$$

Error Optimization – gradient descent

- Learning & Optimization: Assume $J(\theta)$ the objective error function, θ hypothesis parameters.
- Objective: find θ that minimizes $J(\theta)$:
  - update the parameters in the opposite direction of the gradient of the objective function: $\nabla_\theta J(\theta)$ w.r.t. to the parameters
  - Batch gradient descent

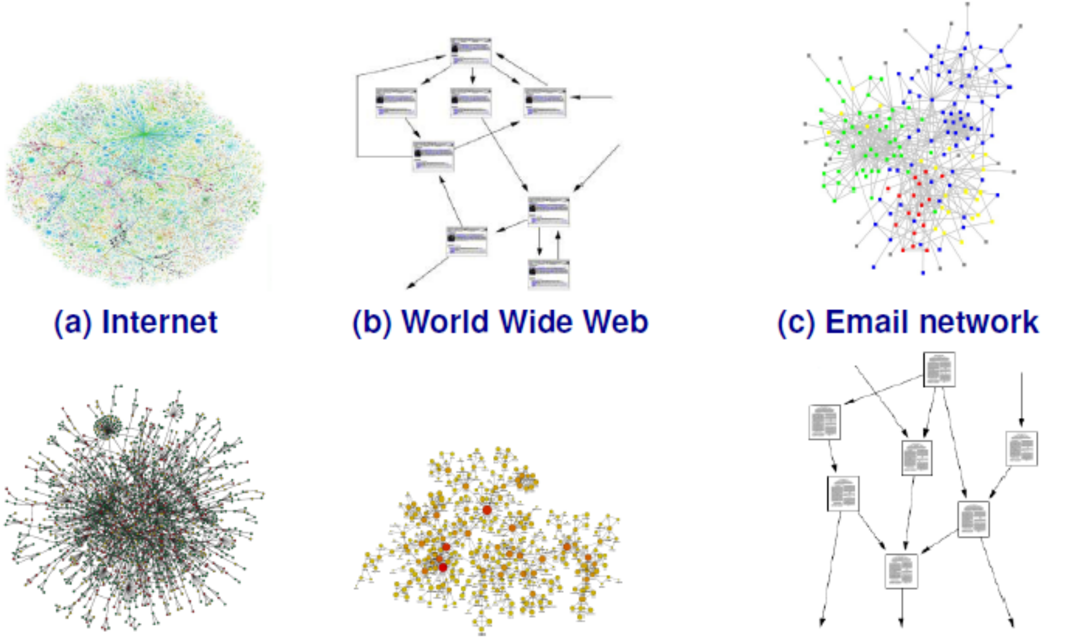
$$\theta = \theta - \eta \nabla_\theta J(\theta)$$

  - η the learning rate
  - *Redundant computations:* as it recomputes gradients for similar examples before each parameter update.

Outline

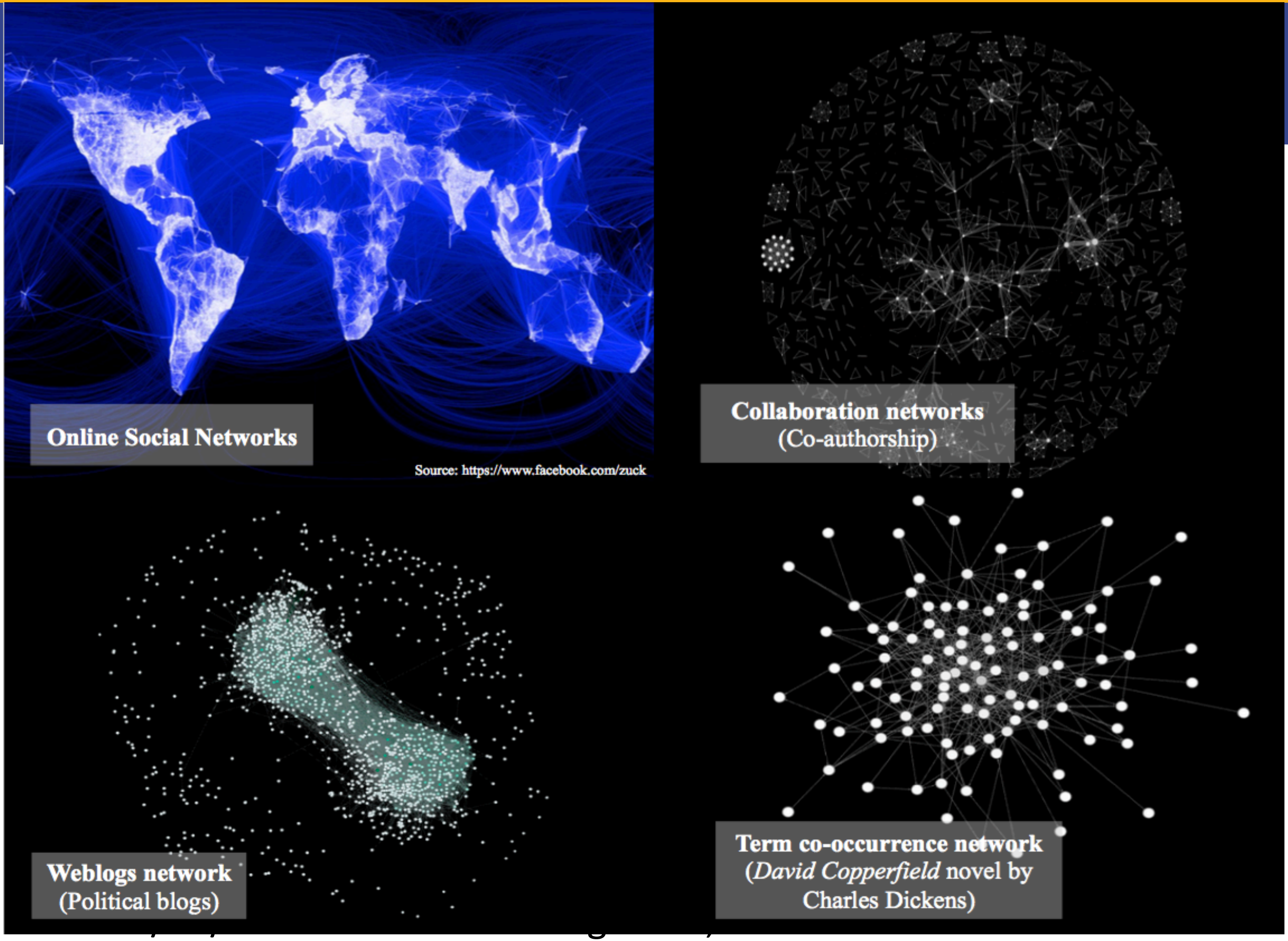
- **Graph Degeneracy**
- Applications
  - Social/Citation networks
  - Text Mining

Graphs are ubiquitous!



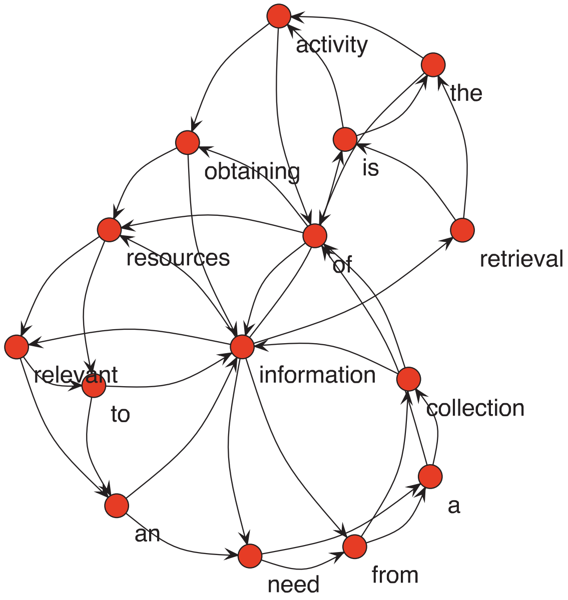
(a) Internet (b) World Wide Web (c) Email network

(d) Protein interactions (e) Collaboration network (f) Citation network


Online Social Networks
Source: https://www.facebook.com/zuck
Collaboration networks
(Co-authorship)
Weblogs network
(Political blogs)
Term co-occurrence network
(David Copperfield novel by
Charles Dickens)

Graph-of-words

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

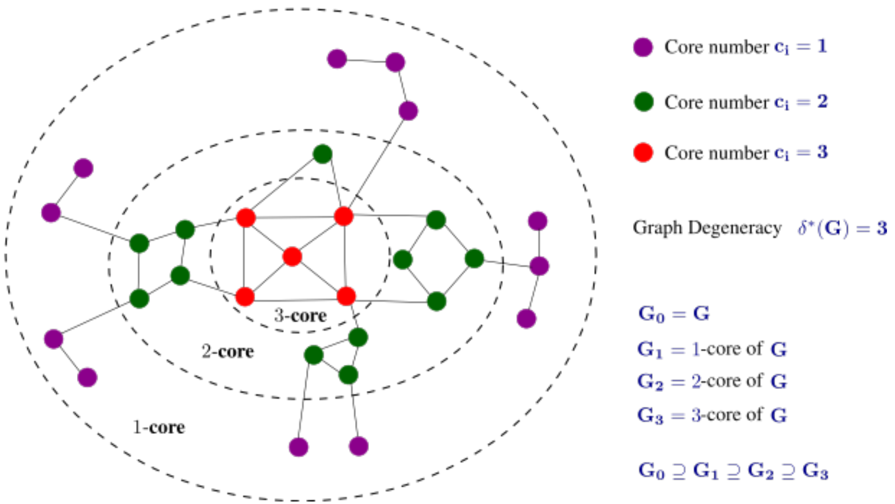


"Graph of word approach for ad-hoc information retrieval", F. Rousseau, M. Vazirgiannis,
Best paper mention award ACM CIKM 2013

Core decomposition in networks

k-Core Decomposition

- Degeneracy for an undirected graph G
  - Also known as the k-core number
  - The k-core of G is the largest subgraph in which every vertex has degree at least k within the subgraph



Also known as graph degeneracy

Important property:
- Fast and easy to compute
- Linear to the size of the graph
- Scalable to large scale graphs

Note:
The degeneracy and the size of the k-core provide a good indication of the cohesiveness of the graph

Algorithm for k-Core Decomposition

Algorithm k-core(G, k)
Input: An undirected graph G and positive integer k
Output: k-core(G)
1. let F := G
2. while there is a node x in F such that $deg_F(x)<k$
   delete node x from F
3. return F

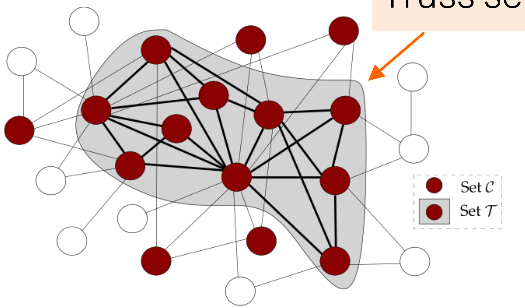
- Many efficient algorithms have been proposed for the computation
  - Time complexity: O(m)

[Batagelj and Zaversnik, ‘03]

K-truss Decomposition (Triangles)

- K-truss decomposition [Cohen '08], [Wang and Cheng '12]
  - Triangle-based extension of the k-core decomposition
  - Each edge of the K-truss subgraph participates in at least K-2 triangles
    - Informally, the "core" of the maximal k-core subgraph
    - Subgraph of higher coherence compared to the k-core

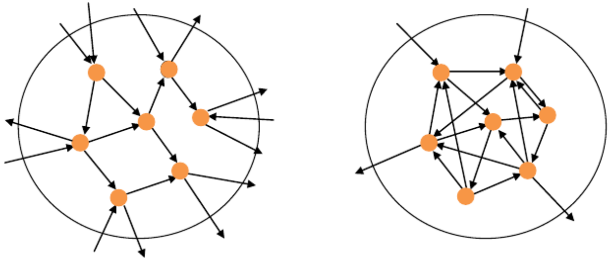
Truss set T



Outline

- Graph Degeneracy
- Applications
  - **Social/Citation networks**
  - Text Mining
- Graph Similarity – Kernels

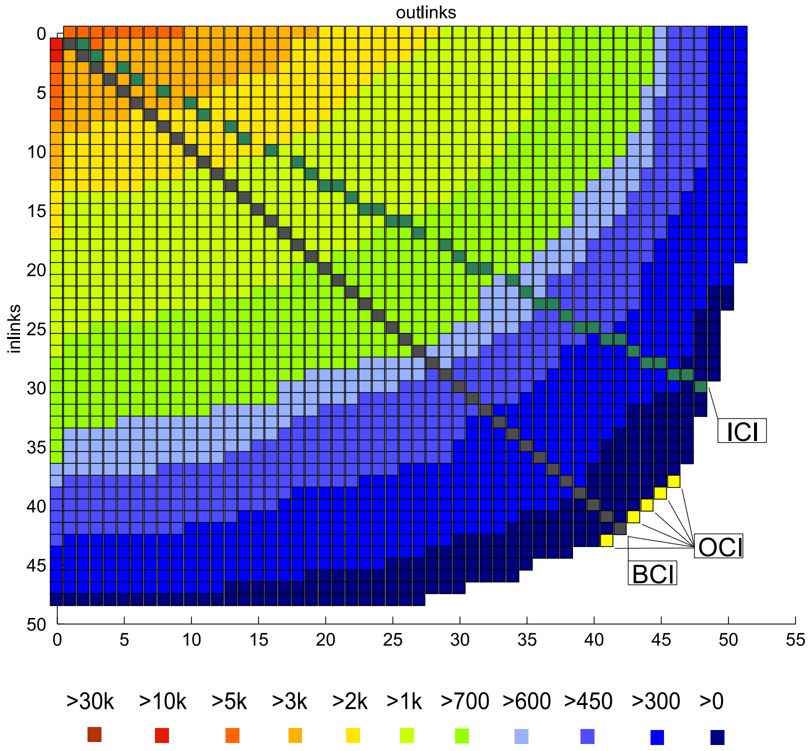
k-core for directed networks [KAIS2014]



- Directed graphs:
  - Wikipedia
  - DBLP – Citation network

- Is there a degeneracy notion for directed graphs?
- We extend the k-core concept in directed graphs by applying a limit on **in/out** edges respectively
  - This provides a **two dimensional range** where cores degenerate
- Trade off between in/out edges can give us a more specific view of the cohesiveness and the “social" behavior

D-core matrix of the DBLP graph



The extreme DBLP D-core authors

Authoritative and Collaborative Scientists

José A. Blakeley
Hector Garcia-Molina
Abraham Silberschatz
Umeshwar Dayal
Eric N. Hanson
Jennifer Widom
Klaus R. Dittrich
Nathan Goodman
Won Kim
Alfons Kemper
Guido Moerkotte
Clement T. Yu
M. Tamer Ã Zsu
Amit P. Sheth
Ming-Chien Shan
Richard T. Snodgrass
David Maier
Michael J. Carey
David J. DeWitt
Joel E. Richardson
Eugene J. Shekita
Waqar Hasan
Marie-Anne Neimat
Darrell Woelk
Roger King
Stanley B. Zdonik
Lawrence A. Rowe
Michael Stonebraker
Serge Abiteboul
Richard Hull
Victor Vianu
Jeffrey D. Ullman
Michael Kifer
Philip A. Bernstein
Vassos Hadzilacos
Elisa Bertino
Stefano Ceri
Georges Gardarin

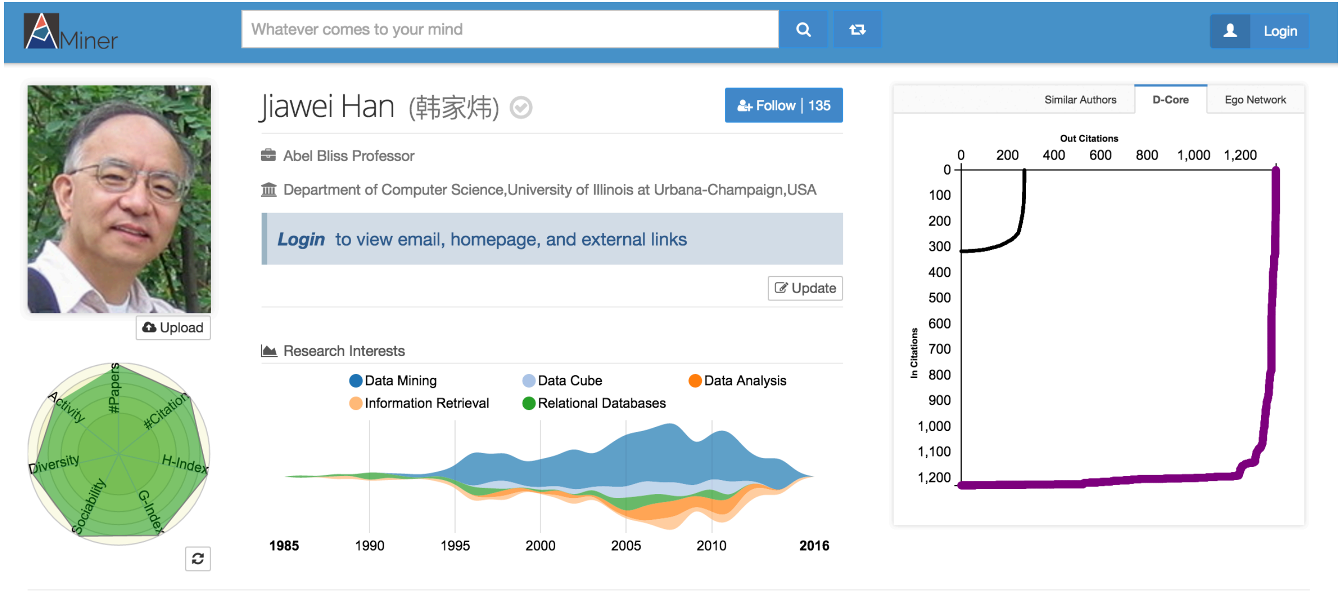
Patrick Valduriez
Ramez Elmasri
Richard R. Muntz
David B. Lomet
Betty Salzberg
Shamkant B. Navathe
Arie Segev
Gio Wiederhold
Witold Litwin
Theo Härder
François Bancilhon
Raghu Ramakrishnan
Michael J. Franklin
Yannis E. Ioannidis
Henry F. Korth
S. Sudarshan
Patrick E. O'Neil
Dennis Shasha
Shamim A. Naqvi
Shalom Tsur
Christos H. Papadimitriou
Georg Lausen
Gerhard Weikum
Kotagiri Ramamohanarao
Maurizio Lenzerini
Domenico Saccà
Giuseppe Pelagatti
Paris C. Kanellakis
Jeffrey Scott Vitter
Letizia Tanca
Sophie Cluet
Timos K. Sellis
Alberto O. Mendelzon
Dennis McLeod
Calton Pu
C. Mohan
Malcolm P. Atkinson
Doron Rotem

Michel E. Adiba
Kyuseok Shim
Goetz Graefe
Jiawei Han
Edward Sciore
Rakesh Agrawal
Carlo Zaniolo
V. S. Subrahmanian
Claude Delobel
Christophe Lecluse
Michel Scholl
Peter C. Lockemann
Peter M. Schwarz
Laura M. Haas
Arnon Rosenthal
Erich J. Neuhold
Hans-Jorg Schek
Dirk Van Gucht
Hamid Pirahesh
Marc H. Scholl
Peter M. G. Apers
Allen Van Gelder
Tomasz Imielinski
Yehoshua Sagiv
Narain H. Gehani
H. V. Jagadish
Eric Simon
Peter Buneman
Dan Suciu
Christos Faloutsos
Donald D. Chamberlin
Setrag Khoshafian
Toby J. Teorey
Randy H. Katz
Miron Livny
Philip S. Yu
Stanley Y. W. Su
Henk M. Blanken

Peter Pistor
Matthias Jarke
Moshe Y. Vardi
Daniel BarbarÃ¡
Uwe Deppisch
H.-Bernhard Paul
Don S. Batory
Marco A. Casanova
Joachim W. Schmidt
Guy M. Lohman
Bruce G. Lindsay
Paul F. Wilms
Z. Meral Özsoyoglu
Gultekin Özsoyoglu
Kyu-Young Whang
Shahram Ghandeharizadeh
Tova Milo
Alon Y. Levy
Georg Gottlob
Johann Christoph Freytag
Klaus Küspert
Louiqa Raschid
John Mylopoulos
Alexander Borgida
Anand Rajaraman
Joseph M. Hellerstein
Masaru Kitsuregawa
Sumit Ganguly
Rudolf Bayer
Raymond T. Ng
Daniela Florescu
Per-Åke Larson
Hongjun Lu
Ravi Krishnamurthy
Arthur M. Keller
Catriel Beeri
Inderpal Singh Mumick
Oded Shmueli

George P. Copeland
Peter Dadam
Susan B. Davidson
Donald Kossmann
Christophe de Maindreville
Yannis Papakonstantinou
Kenneth C. Sevcik
Gabriel M. Kuper
Peter J. Haas
Jeffrey F. Naughton
Nick Roussopoulos
Bernhard Seeger
Georg Walch
R. Erbe
Balakrishna R. Iyer
Ashish Gupta
Praveen Seshadri
Walter Chang
Surajit Chaudhuri
Divesh Srivastava
Kenneth A. Ross
Arun N. Swami
Donovan A. Schneider
S. Seshadri
Edward L. Wimmers
Kenneth Salem
Scott L. Vandenberg
Dallan Quass
Michael V. Mannino
John McPherson
Shaul Dar
Sheldon J. Finkelstein
Leonard D. Shapiro
Anant Jhingran
George Lapis

Adopted by aminer.org



https://aminer.org/

Further resources

Aminer contribution

- https://bitbucket.org/xristosakamad/aminer_dcores/src/master/

Demo – CS

- http://moodle.lix.polytechnique.fr/dcore_demo

Graph degeneracy related papers

- Extensions of graph degeneracy for weighted [ASONAM2011], directed [ICDM2011][KAIS2014][KDD2012], signed [SDM2013] graphs
- Graph degeneracy for clustering [AAAI2014]
- Graph anonymization [KAIS2017][KAIS 2018]
- Influence maximization [WWW2016][Nature/Scientific reports 2016]
- Graph Similarity [IJCAI2018]

Outline

- **Graph Degeneracy**
- Applications
  - Social/Citation networks
  - **Text Mining**

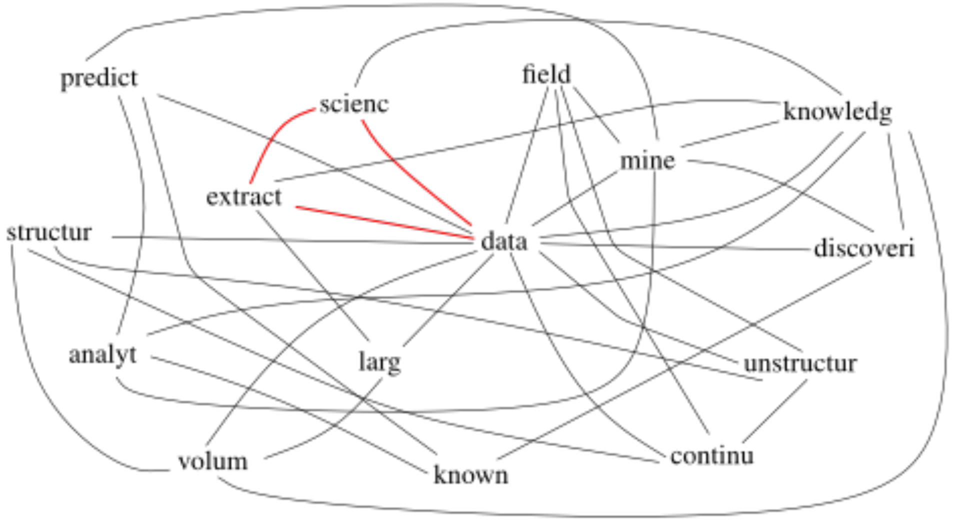
Graph-based text representations

Graph Semantics

- Let $G = (V, E)$ be the graph that corresponds to a document d
- The nodes can correspond to:
  - Paragraphs
  - Sentences
  - Phrases
  - Words [Main focus of the tutorial]
  - Syllables
- The edges of the graph can capture various types of relationships between two nodes:
  - Co-occurrence within a window over the text [Main focus of the tutorial]
  - Syntactic relationship
  - Semantic relationship

Example of Unweighted GoW

Data Science ~~is the~~ extraction of knowledge from large volumes of data that are structured or unstructured which is a continuation of the field of data mining and predictive analytics, also known as knowledge discovery and data mining.

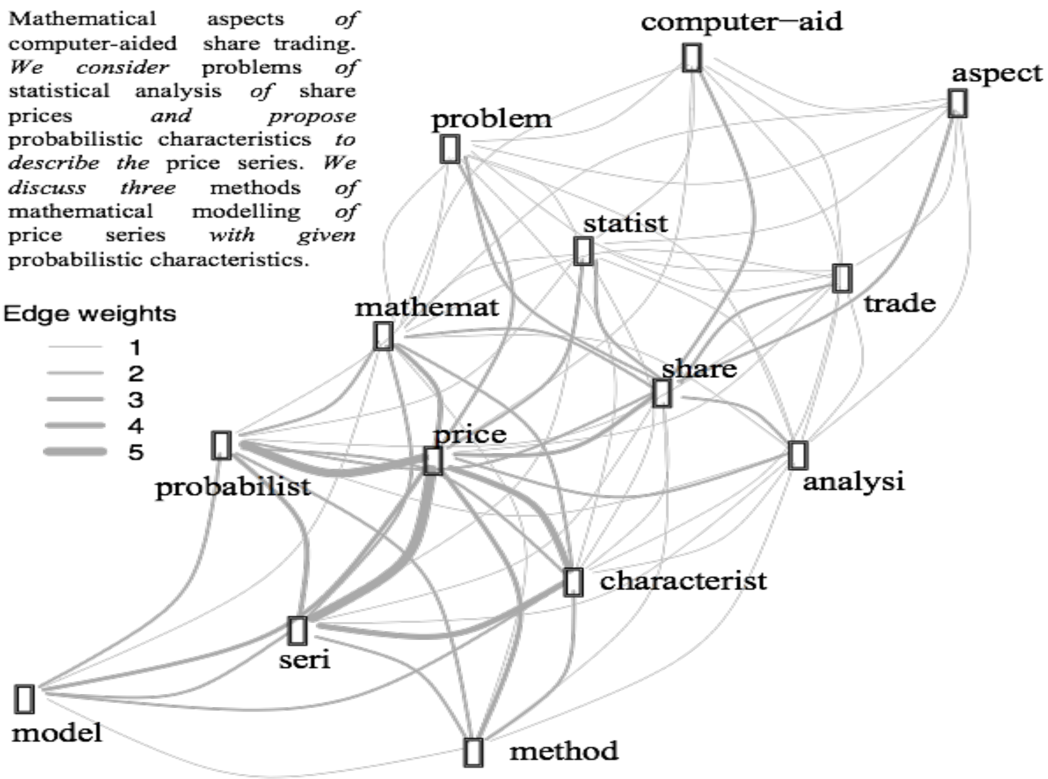


w = 3
unweighted, undirected graph

[Rousseau and Vazirgiannis, CIKM'13 best paper mention award]

Example of Weighed Undirected GoW

Mathematical aspects *of* computer-aided share trading. *We consider* problems *of* statistical analysis *of* share prices *and propose* probabilistic characteristics *to describe the* price series. *We discuss three* methods *of* mathematical modelling *of* price series *with given* probabilistic characteristics.



Single Document Keyword Extraction

Keywords are used everywhere

- Looking up information on the Web (e.g., via a search engine bar)
- Finding similar posts on a blog (e.g., tag cloud)
- For ads matching (e.g., AdWords' keyword planner)
- For research paper indexing and retrieval (e.g., SpringerLink)
- For research paper reviewer assignment

Applications are numerous

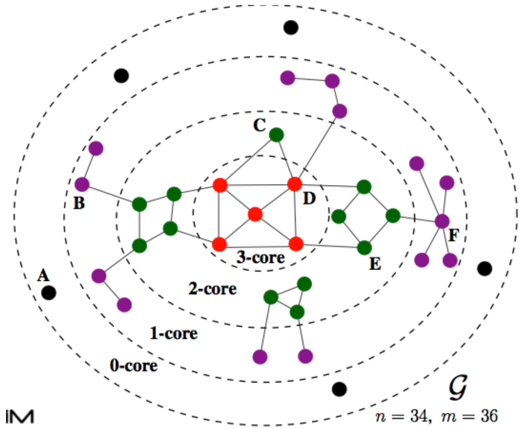
- Summarization (to get a gist of the content of a document)
- Information filtering (to select specific documents of interest)
- Indexing (to answer keyword-based queries)
- Query expansion (using additional keywords from top results)

Graph-based Keyword Extraction (1/2)

Existing graph-based keyword extractors:

- Assign a centrality based score to a node
- Top ranked ones will correspond to the most representative

- TextRank (PageRank) [Mihalcea and Tarau, EMNLP '04]
- HITS [Litvak and Last, MMIES '08]
- Node centrality (degree, betweenness, eigenvector) [Boudin, IJNLP '13]



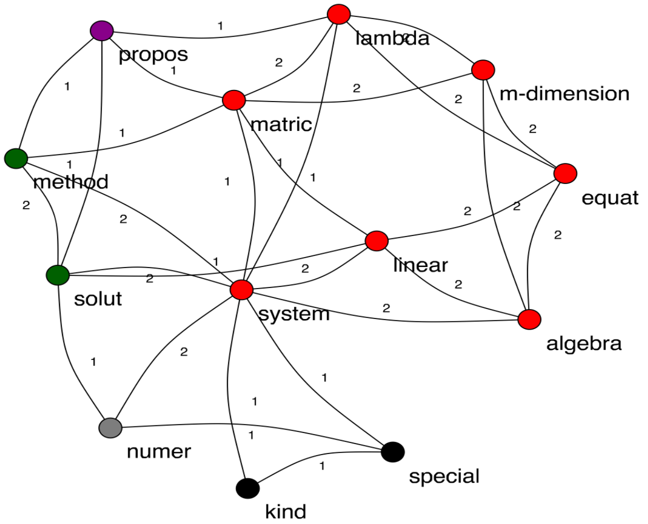
k-core decomposition of the graph

Idea: retain the k-core subgraph of the graph to extract the nodes based on their centrality and cohesiveness

Graph-based Keyword Extraction (2/2)

- Single-document keyword extraction
  - Select the most cohesive sets of words in the graph as keywords
  - Use k-core decomposition to extract the main core of the graph
  - Weighted edges

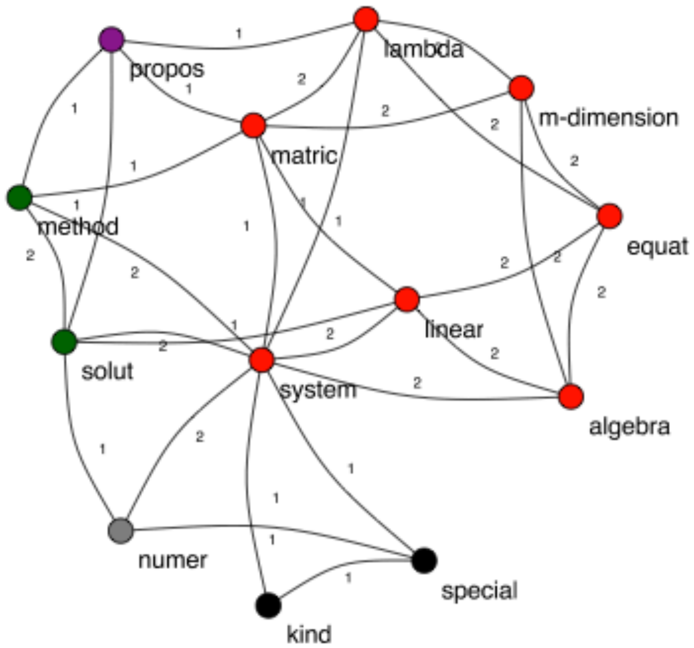
A method for solution of systems of linear algebraic equations with m-dimensional lambda matrices.
A system of linear algebraic equations with m-dimensional lambda matrices is considered. The proposed method of searching for the solution of this system lies in reducing it to a numerical system of a special kind.



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

[Rousseau and Vazirgiannis, ECIR '15]

PageRank vs. k-core



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

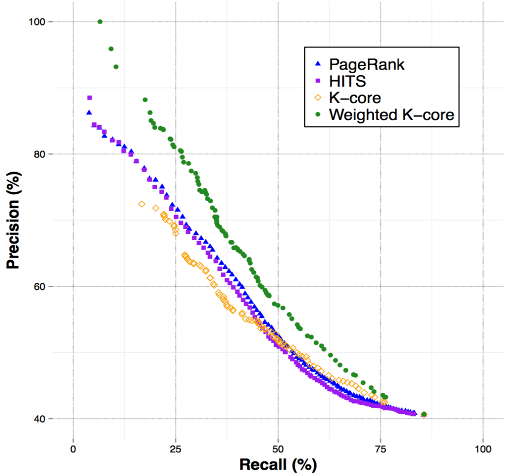
How Many Keywords?

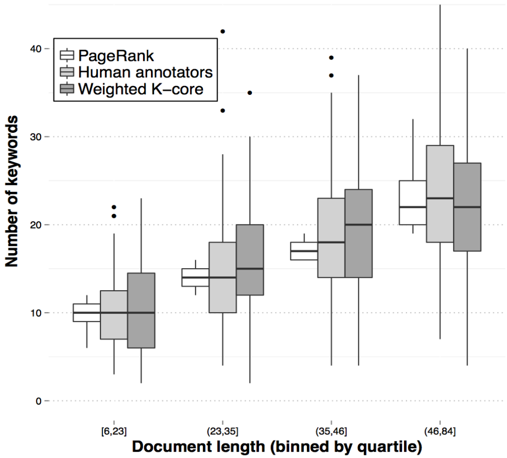
- Most techniques in keyword extraction assign a score to each feature and then take the top ones
- But how many?
  - Absolute number (top X) or relative number (top X%)?
- Besides, at fixed document length, humans may assign more keywords for a document than for another one

X is decided at document level (size of the k-core subgraph) k-cores are adaptive

Performance Evaluation

Precision
Recall
F1-score
Precision/recall

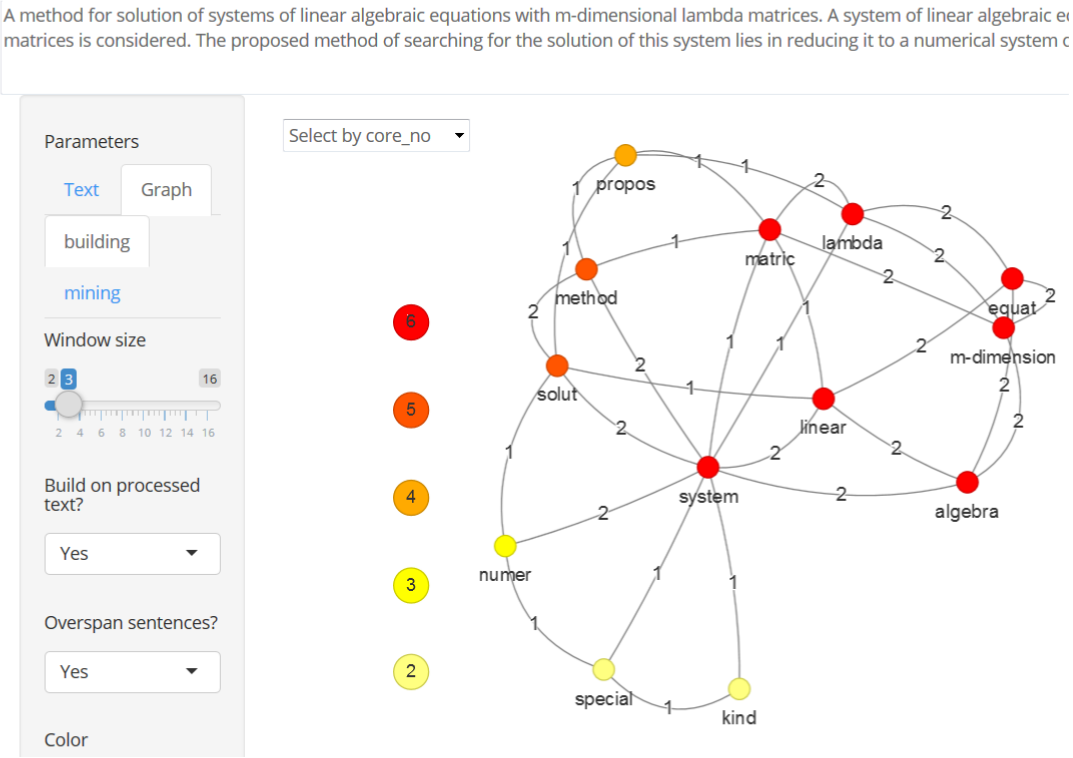




Graph	Dataset	Macro-averaged precision (%)				Macro-averaged recall (%)				Macro-averaged F1-score (%)			
		PageRank	HITS	K-core	WK-core	PageRank	HITS	K-core	WK-core	PageRank	HITS	K-core	WK-core
undirected edges	Hulth2003	58.94	57.86	46.52	**61.24***	42.19	41.80	**62.51***	50.32*	47.32	46.62	49.06*	**51.92***
	Krapi2009	50.23	49.47	40.46	**53.47***	48.78	47.85	**78.36***	50.21	49.59	47.96	46.61	**50.77***
forward edges	Hulth2003	55.80	54.75	42.45	**56.99***	41.98	40.43	**72.87***	46.93*	45.70	45.03	**51.65***	50.59*
	Krapi2009	47.78	47.03	39.82	**52.19***	44.91	44.19	**79.06***	45.67	45.72	44.95	46.03	**47.01***
backward edges	Hulth2003	59.27	56.41	40.89	**60.24***	42.67	40.66	**70.57***	49.91*	47.57	45.37	45.20	**50.03***
	Krapi2009	51.43	49.11	39.17	**52.14***	49.96	47.00	**77.60***	50.16	**50.51**	47.38	46.93	50.42

GoWvis visualization tool

GoWvis Visualization Tool



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

[Tixier et al., ACL ‘16]

GoWvis

- Builds a graph-of-words and displays an interactive representation of any text pasted by the user
- Allows the user to tune many parameters:
  - Text pre-processing (stopword removal, …)
  - Graph building (window size, …)
  - Graph mining (node ranking and community detection algorithms, …)
- Extracts keyphrases and generates a summary of the input text
- Built in R Shiny with the visNetwork library

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

Other efforts with GoW for NLP

- Extractive summarization [EACL2017][ACL2018]
- Event Detection in twitter streams [ECIR2018][AAAI-ICWSM2015]

Outline

- Graph Degeneracy
- Applications
  - Social/Citation networks
  - Text Mining
- **Graph Similarity – Kernels**

GraKeL - Python package extension for graph similarity

- GraKeL is a Python package extension for graph kernels.
- Project is currently under alpha development stage and is uploaded on pypi-test.
- Code: https://github.com/ysig/GraKeL/tree/develop.
  Documentation: https://ysig.github.io/GraKeL/dev/.
  Paper: https://arxiv.org/abs/1806.02193.

Implemented kernels in Grakel

- Core Kernel Framework
- Edge Histogram Kernel
- Graph Hopper Kernel
- Graphlet Sampling Kernel
- Hadamard Code Kernel
- Kernel (general class)
- Lovasz Theta Kernel
- Multiscale Laplacian Kernel
- Neighborhood Hash Kernel
- Neighborhood Subgraph Pairwise Distance Kernel
- ODD-STh Kernel
- The Propagation Kernel
- Pyramid Match Kernel
- Random Walk Kernel
- Shortest Path Kernel
- Subgraph Matching Kernel
- SVM Theta Kernel
- Vertex Histogram Kernel
- Weisfeiler Lehman Framework

Thank You! - Questions?

Credits to my collaborators

- Dr.C. Giatsidis, (X)
- Prof. Malliaros (Ec. Centrale, Paris)
- Dr. F. Rousseau (Google)
- Dr. G. Nikolentzos (X)
- Prof. D. Thilikos (CNRS)
- Dr. M. Rossi (X)
- P. Meladianos (AUEB)
- …

Michalis Vazirgiannis
Data Science and Mining group, École Polytechnique
http://www.lix.polytechnique.fr/~mvazirg
https://www.lix.polytechnique.fr/dascim/
Twitter: @mvazirg