Graph-based Methods in Pattern Recognition and Document Image Analysis (GMPRDIA)

Centre de Visió per Computador

Informatique Image Interaction
Univ. La Rochelle

Tutorial at the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR2017)
Graph-based Methods in Pattern Recognition and Document Image Analysis (GMPRDIA)
http://gmprdia.univ-lr.fr

Sunday November 12th 2017

Session-1 (9h00 - 10h30)

1. Introduction (15m)
2. Graph representation (25m)
3. Graph matching / edit-distance (20m)
4. Graph embedding / Graph kernel (30m)

Coffee Break 10h30-11h00

Session-2 (11h - 12h30)

1. Graph indexing, graph retrieval, subgraph spotting and graph diffusion, graph serialization (20m)
2. Neural network on graphs (30m)
3. Programming languages, evaluation protocols, datasets and Programming Hands-on: Graph classification with RW kernel (25m)
4. Discussion (12h15- 12h30)

Session-1 (9h00 - 10h30)

1. Introduction
2. Graph representation
3. Graph matching / edit-distance
4. Graph embedding / Graph kernel

Introduction

Tutorial organizers

Scientific Committee
Prof. Josep LLADOS CANET (1)
Prof. Jean-Marc OGIER (2)

Organizing Committee and Speakers
Dr. Anjan DUTTA (1)
Dr. Muhammad Muzzamil LUQMAN (2)

(1) CVC Barcelona, Spain
(2) L3i La Rochelle, France

Tutorial organizers

Organizing Committee and Speakers
Dr. Anjan DUTTA
CVC Barcelona, Spain

- Marie-Curie postdoctoral fellow under the P-SPHERE project.
- Ph.D. in Computer Science from the Universitat Autònoma de Barcelona (UAB) in the year of 2014.
- Ph.D. thesis titled “Inexact Subgraph Matching Applied to Symbol Spotting in Graphical Documents”
- Research interests
    - graph-based representation for visual objects
    - graph-based algorithms for solving various tasks in Computer Vision, Pattern Recognition and Machine Learning.

Tutorial organizers

Organizing Committee and Speakers
Dr. Muhammad Muzzamil LUQMAN
L3i La Rochelle, France

- Research Scientist (Permanent)
- Ph.D. in Computer Science from François Rabelais University of Tours (France) and Autonoma University of Barcelona (Spain).
- Ph.D. thesis titles “Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images”.
- Research interests
  - Structural Pattern Recognition
  - Document Image Analysis
  - Camera-Based Document Analysis and Recognition
  - Graphics Recognition
  - Machine Learning

Introduction - GMPRDIA

	Pattern Recognition	
	Structural	Statistical
Data structure	symbolic data structure	numeric feature vector
Representational strength	Yes	No
Fixed dimensionality	No	Yes
Sensitivity to noise	Yes	No
Efficient computational tools	No	Yes

Introduction - GMPRDIA

Goals

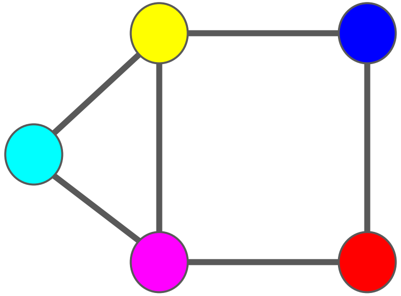
- A quick overview of graph-based representations and graph-based methods for pattern recognition and document image analysis

- Current trends in graph-based methods for pattern recognition and document image analysis

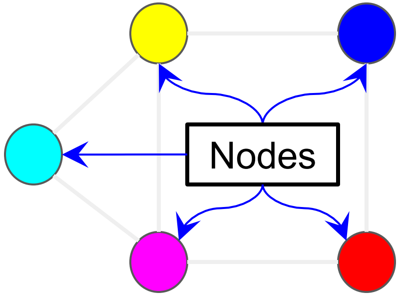
Graph representation

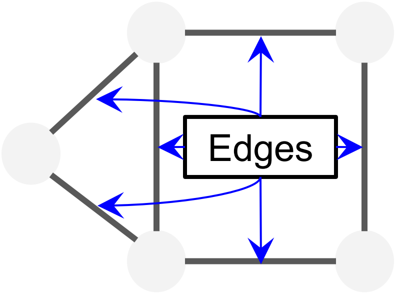
Graph

- A *graph* $G = (V, E)$ is a mathematical structure for representing relationships.

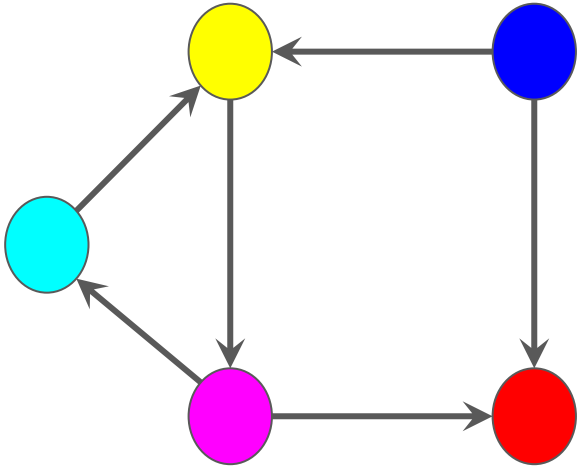


- A graph $G = (V, E)$ consists of a set of *nodes* V connected by *edges* E.

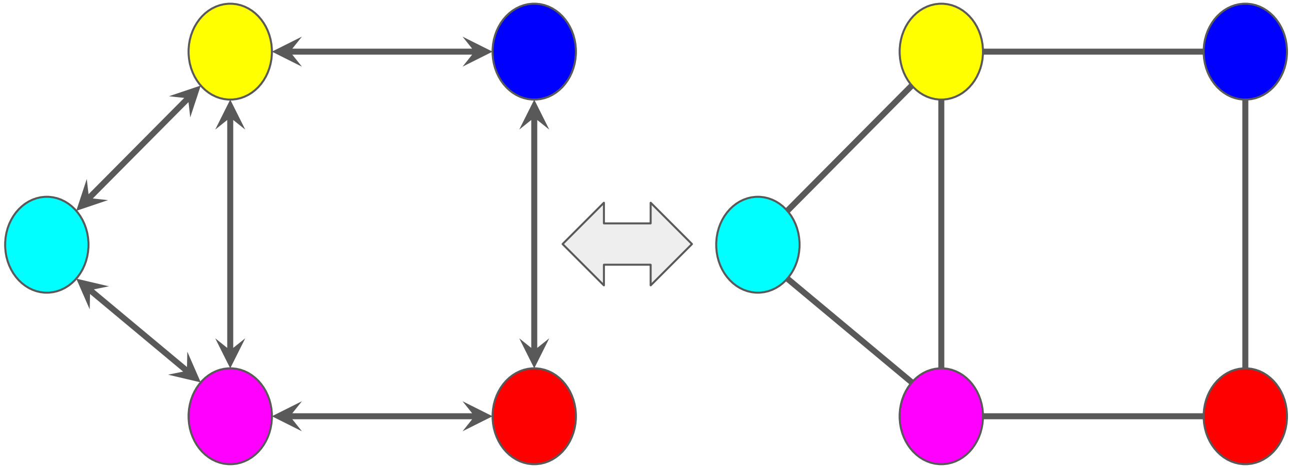




Directed and Undirected Graph



Directed Graph

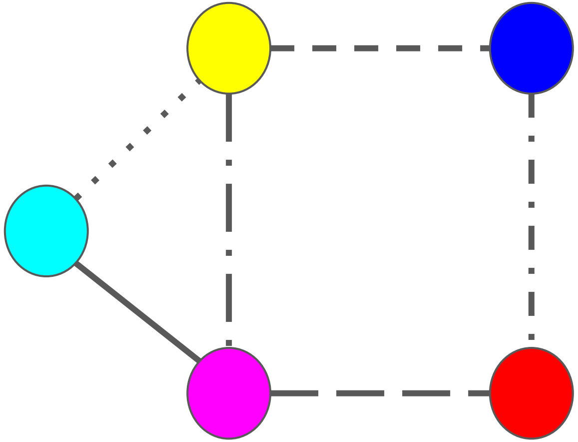


Undirected Graph

Attributed Graph

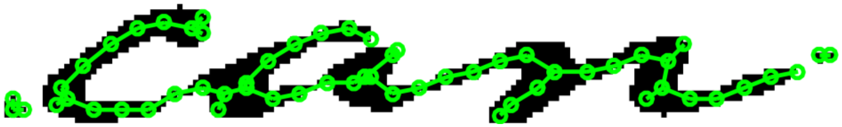
An attributed Graph is a 4-tuple $G = (V, E, \alpha, \beta)$

- Set of nodes V
- Set of edges $E \subseteq V \times V$
- Node attribute function $\alpha : V \to L_V$
- Edge attribute function $\beta : E \to L_E$



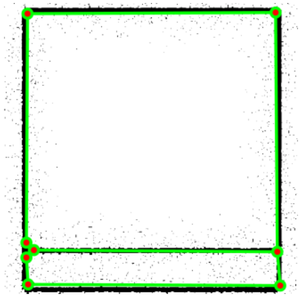
Graph Representation: Examples

- Critical points, grid etc as nodes.
- Adjacent nodes on the writing are joined.
- Normalized coordinates as node attributes



Histograph dataset
(http://www.histograph.ch/)

- Critical points as nodes.
- Adjacent nodes on the symbol are joined.
- Coordinate as node attributes.
- Line type as edge attributes.



GREC dataset
(http://www.fki.inf.unibe.ch/databases)

Graph Representation: Issues to Consider

Graph representation of objects depends on:

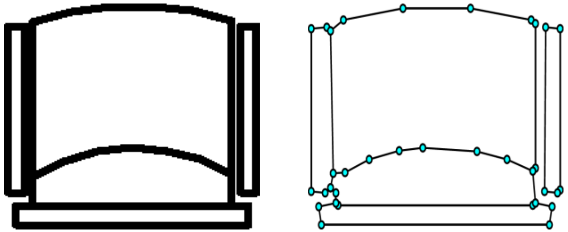
1. Problem definition
2. Type of solution / methodology
3. Stability and noise tolerance

Discriminant units of information in an underlying image for representing it by a graph

- Critical Points
- Line Segments
- Homogeneous Regions
- Keypoints
- Convex Regions
- etc.

Critical Points

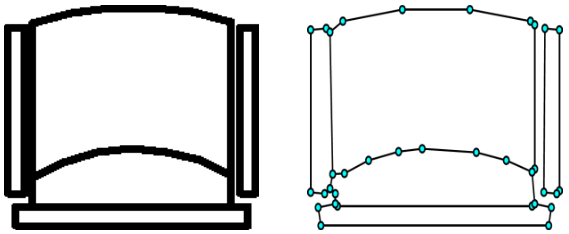
- Critical points from skeleton or edge analysis as nodes.
- Type of edges:
  - Adjacency
  - Proximity
  - k-NN
  - Delaunay triangulation
- Example
  - Symbol spotting by hashing serialized subgraphs.
  - Critical points as nodes and their connections as edges.



A. Dutta, J. Lladós, and U. Pal. A symbol spotting approach in graphical documents by hashing serialized graphs. In PR, vol. 46, no. 3, pp. 752-768, 2013.

Line Segments

- Line segments from skeleton or edge analysis as nodes.
- Type of edges:
  - Adjacency
  - Proximity
  - k-NN
  - Delaunay triangulation
- Example
  - Subgraph matching applied to symbol spotting.
  - Each line segment as a node and upto 3 nearest neighbors are joined to form edges.



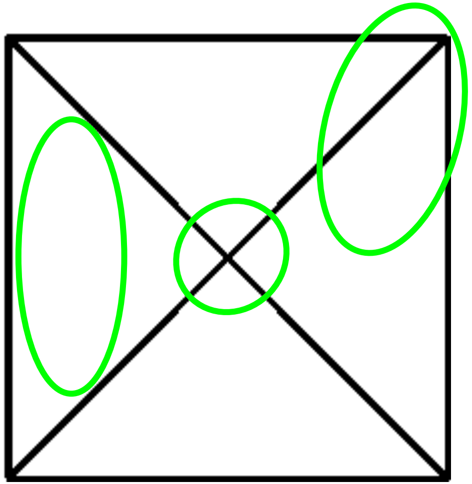
A. Dutta, J. Lladós, H. Bunke and U. Pal. “A Product graph based method for dual subgraph matching applied to symbol spotting”. GREC, 2014.

Homogeneous Regions

- Regions either existing or generated by a preprocessing stage as nodes.
- Type of edges:
  - Adjacency
  - Proximity
  - Delaunay triangulation
- Example
  - SSGCI competition, ICPR, 2016.
  - RAG of cartoon characters.
  - Subgraph spotting.

Keypoints

- Detected keypoints using some off-the-shelf algorithm as nodes.
- Type of edges:
  - Proximity
  - k-NN
  - Delaunay triangulation
- Example
  - Symbol recognition.
  - Shape context of detected SIFT interest points.



T. H. Do, S. Tabbone, O. R. Terrades. “Sparse representation over learned dictionary for symbol recognition". SP, pp. 36-47, 2016.

Example: Skeleton Graph

- Skeleton graph.
- Each junction or end point as a node of the graph.
- Edges are created following the skeleton.

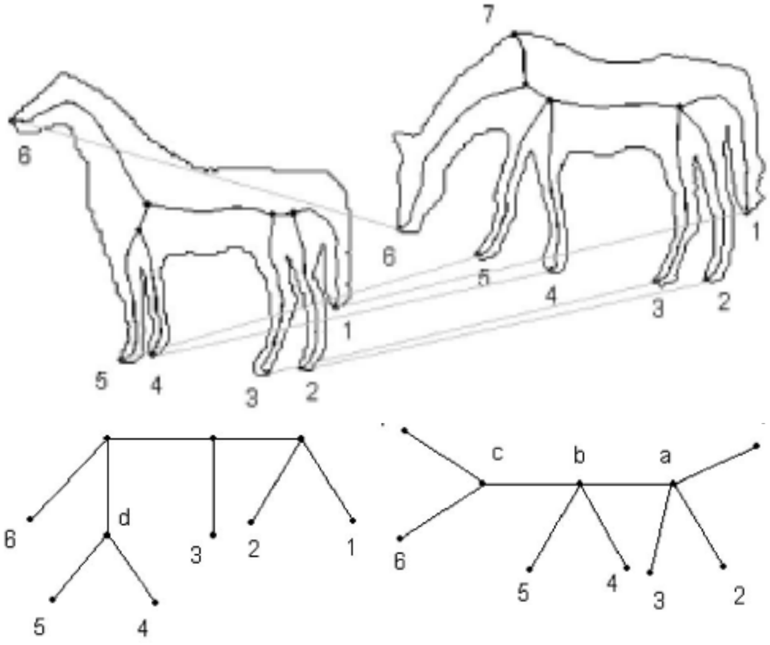


Figure credit: Bai and Latecki PAMI 2008

X. Bai and L. J. Latecki. Path Similarity Skeleton Graph Matching. IEEE TPAMI, vol. 30, no. 7, 1282-1292, 2008.

Example: Region Adjacency Graph

- Region adjacency graph.
- Each white region as a node in the graph.
- Each pair of adjacent nodes is connected by an edge.

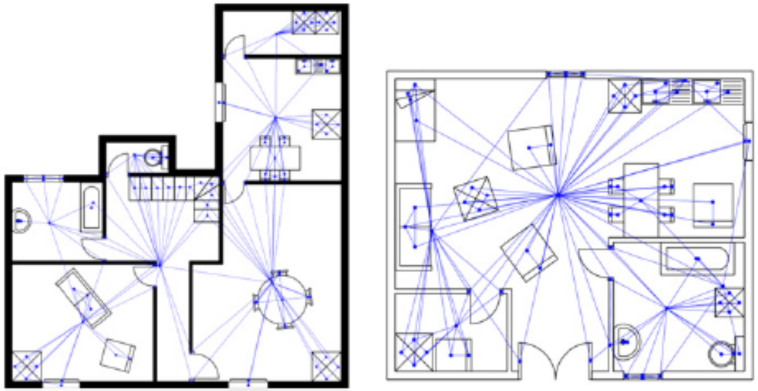


Figure credit: Le Bodic et al 2012

P. L. Bodic, P. Héroux, S. Adam and Y. Lecourtier. An integer linear program for substitution-tolerant subgraph isomorphism and its use for symbol spotting in technical drawings. PR, vol. 45, no. 12, pp. 4214-4224, 2012.

Example: Graph of convexities

- Convex part segmentation.
- Each convex part as node.
- Nearest nodes are joined as edges.

Figure credit: Riba et al, PRL 2017

P. Riba, J. Lladós, A. Fornés, A. Dutta. Large-scale graph indexing using binary embeddings of node contexts for information spotting in document image databases. PRL, vol. 87, pp. 203 - 211, 2017.

Learning Graph Representation

- Learning graph that best represent an image for matching to another relevant image.
- Fully connected graph of detected key points.
- Learning node and edge parameters that prioritize a set of nodes for a particular structure.

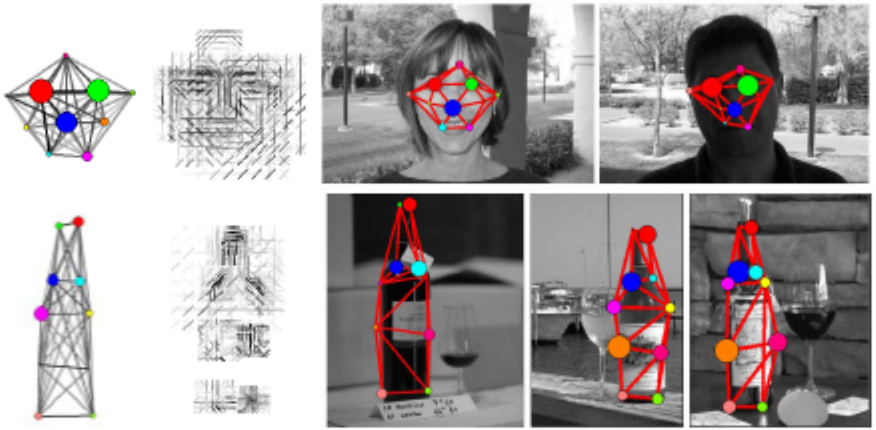
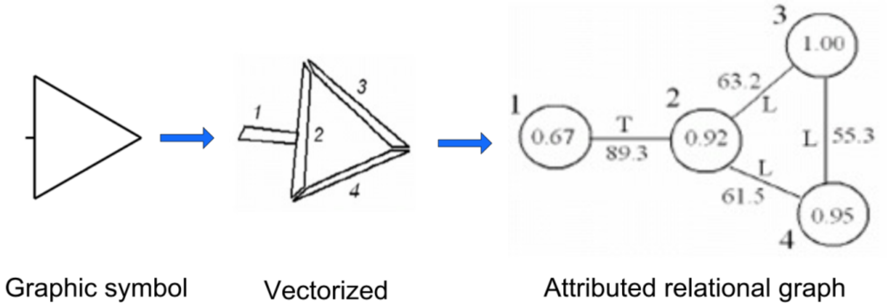


Figure credit: Cho et al 2013

M. Cho, K. Alahari and J. Ponce. Learning Graphs to Match. ICCV, 2013.

Example: Vecto-Quad graph representation

- Graph representation developed for line drawings
- Each node in the graph represents a line in underlying image
- Thin lines are termed as vectors
- Thick lines or filled shapes are termed as quadrilaterals
- Connections between the vectors/quadrilaterals are represented by edges
- Attributes on nodes as well as edges

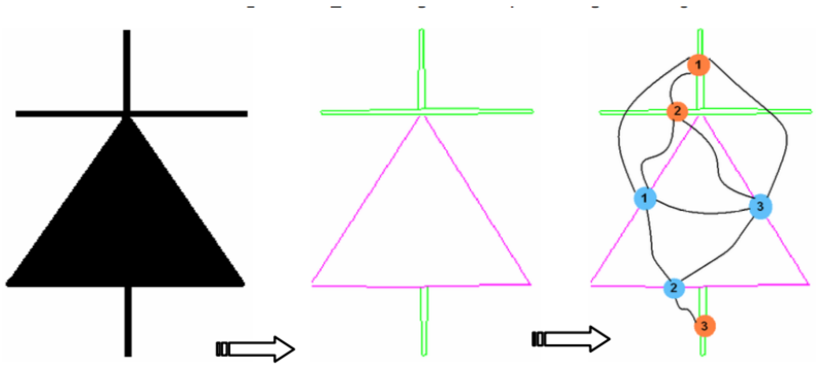


R. Qureshi, J. Ramel, H. Cardot, and P. Mukherji, “Combination of symbolic and statistical features for symbols recognition,” in IEEE ICSCN, 2007, pp. 477–482.

J.Y. Ramel, N. Vincent, H. Emptoz, "A structural Representation for understanding line-drawing images", InternationalJournalonDocumentAnalysisandRecognition, vol.3(2),2000,pp.58- 66.

Example: Vecto-Quad graph representation

- Vectors and Quadrilaterals representation well adapted to the underlying line-drawing images

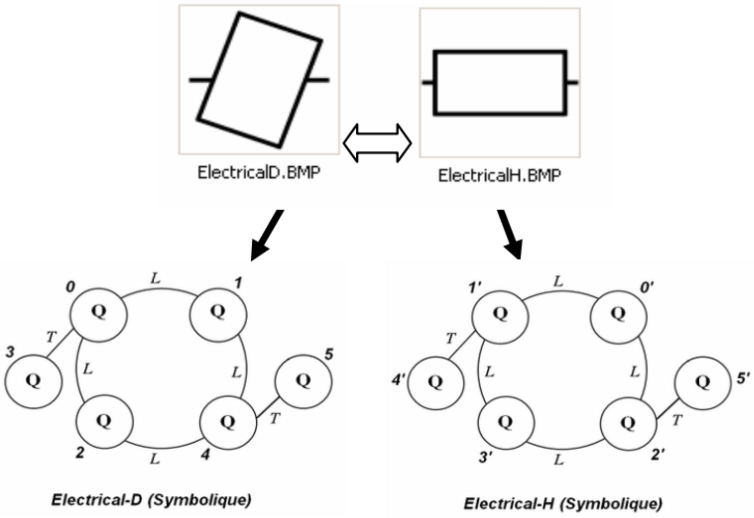


R. Qureshi, J. Ramel, H. Cardot, and P. Mukherji, “Combination of symbolic and statistical features for symbols recognition,” in IEEE ICSCN, 2007, pp. 477–482.

J.Y. Ramel, N. Vincent, H. Emptoz, "A structural Representation for understanding line-drawing images", InternationalJournalonDocumentAnalysisandRecognition, vol.3(2),2000,pp.58- 66.

Example: Vecto-Quad graph representation

- Graph-based representations have built-in rotation invariance

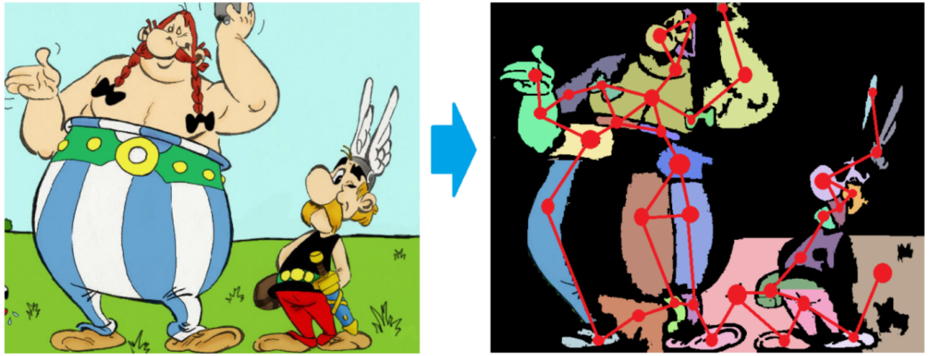


R. Qureshi, J. Ramel, H. Cardot, and P. Mukherji, “Combination of symbolic and statistical features for symbols recognition,” in IEEE ICSCN, 2007, pp. 477–482.

J.Y. Ramel, N. Vincent, H. Emptoz, "A structural Representation for understanding line-drawing images", InternationalJournalonDocumentAnalysisandRecognition, vol.3(2),2000,pp.58- 66.

Example: MSER-regions based graph representation

- Graph representation developed for colored comic images
- Each node in graph represents an MSER region in underlying image
- Spatial relations between MSER regions are represented by edges in graph
- Attributes on nodes as well as edges

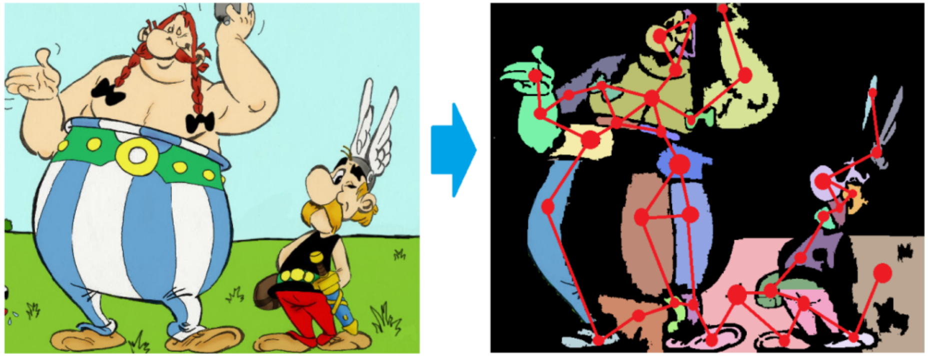


Thanh-Nam Le, Muhammad Muzzamil Luqman, Jean-Christophe Burie, Jean-Marc Ogier: Content-based comic retrieval using multilayer graph representation and frequent graph mining. ICDAR 2015: 761-765

M. M. Luqman, H. N. Ho, J.-c. Burie, and J.-M. Ogier, "Automatic indexing of comic page images for query by example based focused content retrieval," in 10th 1APR International Workshop on Graphics Recognition, United States, Aug. 2013.

Example: MSER-regions based graph representation

- Multilayer graph representation
    - Color layer
    - Hu-moments layer
    - Compactness layer

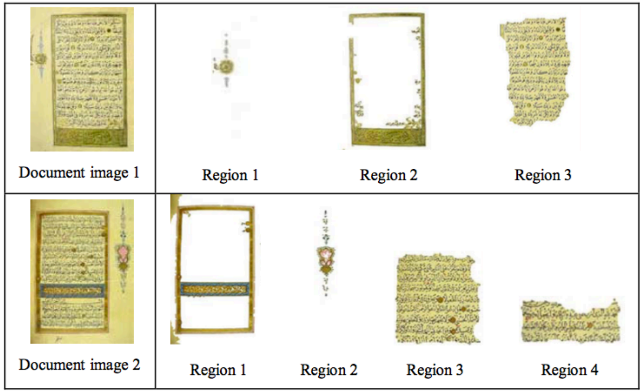


Thanh-Nam Le, Muhammad Muzzamil Luqman, Jean-Christophe Burie, Jean-Marc Ogier: Content-based comic retrieval using multilayer graph representation and frequent graph mining. ICDAR 2015: 761-765

M. M. Luqman, H. N. Ho, J.-c. Burie, and J.-M. Ogier, "Automatic indexing of comic page images for query by example based focused content retrieval," in 10th 1APR International Workshop on Graphics Recognition, United States, Aug. 2013.

Example: Fuzzy Attributed Relational Graphs (FARG)

- segmentation errors may occur in document images: (noise and degradation, overlapping layouts, presence of handwriting, etc
- Therefore, representing the content by fuzzy graphs allows to capture the maximum information from a document image with a certain error-tolerance.
- Structural and visual features represented by fuzzy concepts, such as “Near” and “Far”, “Big” and “Small”, etc.



Ramzi Chaieb, Karim Kalti, Muhammad Muzzamil Luqman, Mickaël Coustaty, Jean-Marc Ogier, Najoua Essoukri Ben Amara: Fuzzy generalized median graphs computation: Application to content-based document retrieval. Pattern Recognition 72: 266-284 (2017)

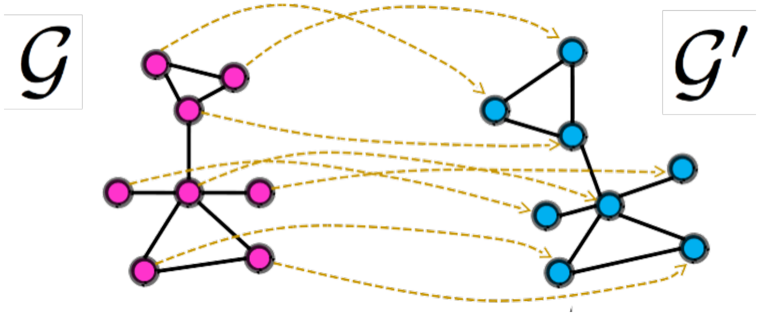
Summary: Graph representation

- What is a graph representation?
- What are important constituent parts of graph-based representations?
- What are some of the possible discriminant units of information in an underlying image for constructing graph-based representation of it?
- Some example graph-based representations, used in PR and DIA works

Graph matching

Graph matching

Finding matches between two graphs.

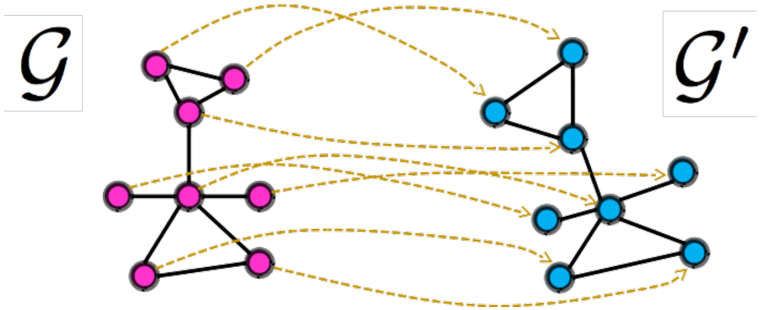


$$\mathbf{X} \in \{0,1\}^{nn'}$$

- $\mathbf{X}_{ia} = 1$ if node i in G corresponds to node a in G'
- $\mathbf{X}_{ia} = 0$ otherwise

Graph matching

Maximizing the matching score S

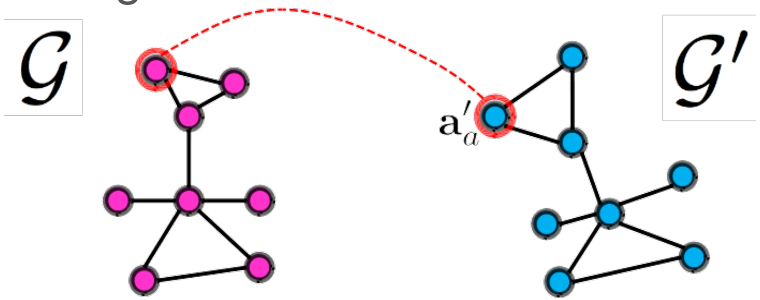


$$\mathbf{X}^* = \arg\max_{\mathbf{X}} S(\mathcal{G}, \mathcal{G}', \mathbf{X})$$

$$\text{s.t.} \begin{cases} \mathbf{X} \in \{0,1\}^{nn'} \\ \sum_{i=1}^{n} \mathbf{X}_{ia} \leq 1, \sum_{a=1}^{n'} \mathbf{X}_{ia} \leq 1 \end{cases}$$

Graph matching

How to measure the matching score S?

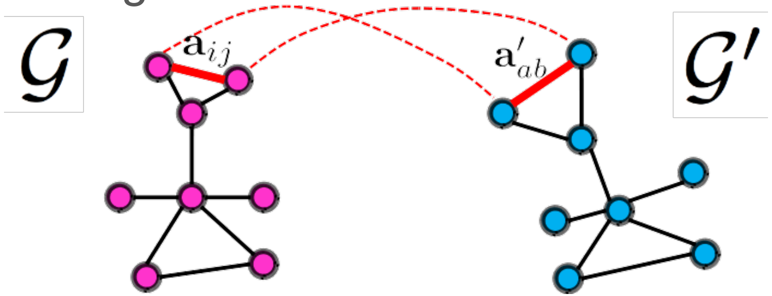


$$S_V(\mathbf{a}_i, \mathbf{a}'_a)$$

- Each node and each edge has its own attribute
- Node similarity function

Graph matching

How to measure the matching score S ?

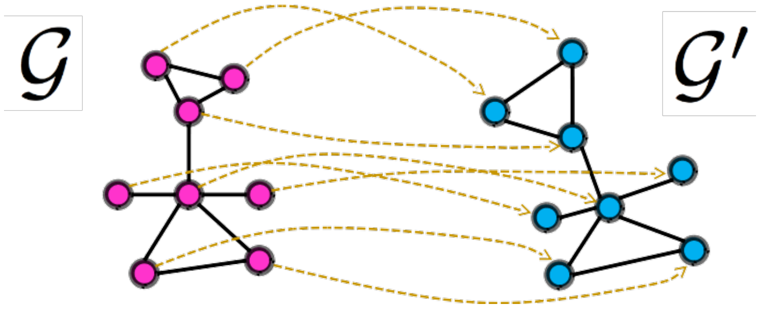


$$S(\mathcal{G}, \mathcal{G}', \mathbf{X}) = \sum_{\mathbf{X}_{ia}=1} S_V(\mathbf{a}_i, \mathbf{a}'_a) + \sum_{\mathbf{X}_{ia}=1, \mathbf{X}_{jb}=1} S_E(\mathbf{a}_{ij}, \mathbf{a}'_{ab})$$

- Sum of S_V and S_E values for the assignment $\mathbf{X}$.

Graph matching

How to measure the matching score S?



- $\mathbf{X}_{\mathrm{ia}} = 1$ if node i in $\mathcal{G}$ corresponds to node a in $\mathcal{G}'$
- $\mathbf{X}_{\mathrm{ia}} = 0$ otherwise

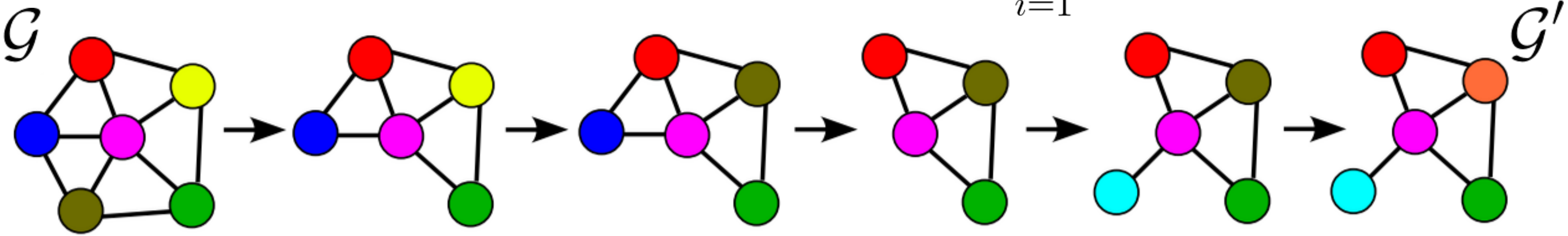
Advances in graph matching

- Quadratic assignment problem
  - NP-hard, thus exact solution is infeasible
- Advances in approximate algorithms
  - Relaxation and Projection
- Graph edit distance
- Other approximate algorithms
  - Spectral decomposition
  - Semidefinite programming
  - Continuous relaxation

Graph edit distance

- A measure of similarity between two graphs.
- Node and edge insertion, deletion, substitution.
- Summation of the edit costs

$$\mathrm{GED}(\mathcal{G}, \mathcal{G}') = \min_{(e_1,\dots,e_k)\in\mathcal{P}(\mathcal{G},\mathcal{G}')} \sum_{i=1}^{k} c(e_i)$$



A. Sanfeliu, K. S. Fu. A distance measure between attributed relational graphs for pattern recognition. IEEE TSMC, vol. 13, no. 3, 1983

Spectral decomposition

- Estimate graph permutation matrix $\mathbf{X}$ as an orthogonal one, i.e., $\mathbf{X}^T\mathbf{X}=\mathbf{I}$
- Under this constraint, GM can be solved as a closed form of eigen-value problem.
- Further relaxation by constraining $\mathbf{X}$ to be unit length, i.e., $\|\mathrm{vec}(\mathbf{X})\|_2 = 1$

1. T. Caelli and S. Kosinov, “An eigenspace projection clustering method for inexact graph matching”, IEEE TPAMI, vol. 26, no. 4, pp. 515–519, 2004.
2. M. Leordeanu and M. Hebert, “A spectral technique for correspondence problems using pairwise constraints”, ICCV, 2005.
3. T. Cour, P. Srinivasan, and J. Shi, “Balanced graph matching”, NIPS, 2006.

Semidefinite programming

- Approximate the non-convex constraint $\mathbf{Y}=\mathrm{vec}(\mathbf{X})\mathrm{vec}(\mathbf{X})^T$ as $\mathbf{Y}-\mathrm{vec}(\mathbf{X})\mathrm{vec}(\mathbf{X})^T \geq 0$ where $\mathbf{Y} \in \mathbb{R}^{nn' \times nn'}$
- Having $\mathbf{Y}$, $\mathbf{X}$ can be approximated by a randomized algorithm.
- Theoretical guarantee to find a polynomial time approximation.
- Practically expensive as the variable $\mathbf{Y}$ squares the problem size.

1. H. S. Torr, “Solving Markov random fields using semidefinite programming”, AISTATS, 2003.
2. C. Schellewald and C. Schnörr, “Probabilistic subgraph matching based on convex relaxation”, EMMCVPR, 2005.

Continuous relaxation

- Estimates $\mathbf{X}$ in the convex hull of the set of permutation matrices

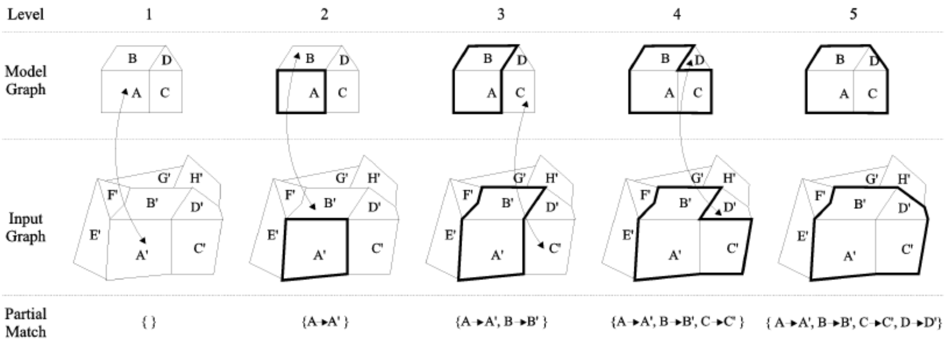
$$\mathcal{D} = \{\mathbf{X}|\mathbf{X} \in [0,1]^{nn'}, \mathbf{X}\mathbf{1}_{n'} \leq \mathbf{1}_n, \mathbf{X}^T\mathbf{1}_n \leq \mathbf{1}_{n'}\}$$

- Doubly stochastic relaxation.
- Non-convex quadratic assignment problem.

1. H. A. Almohamad and S. O. Duffuaa, “A linear programming approach for the weighted graph matching problem,” IEEE TPAMI, vol. 15, no. 5, pp. 522–525, 1993.
2. S. Gold and A. Rangarajan, “A graduated assignment algorithm for graph matching,” IEEE TPAMI, vol. 18, no. 4, pp. 377–388, 1996.
3. B. J. van Wyk and M. A. van Wyk, “A POCS-based graph matching algorithm,” IEEE TPAMI, vol. 26, no. 11, pp. 1526–1530, 2004.
4. L. Torresani, V. Kolmogorov, and C. Rother, “Feature correspondence via graph matching: Models and global optimization”, ECCV, 2008.
5. M. Cho, J. Lee, and K. M. Lee, “Reweighted random walks for graph matching”, ECCV, 2010.
6. F. Zhou and F. De la Torre, “Factorized graph matching”, IEEE TPAMI, vol. 38, no. 9, 2016.

GM in Document Analysis: Example 1

Symbol Recognition by Error-Tolerant Subgraph Matching



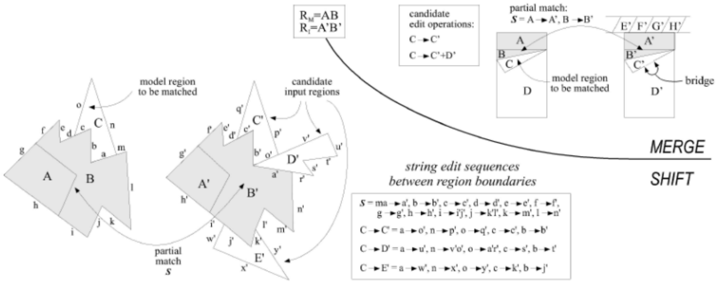


Figure credit: Lladós et al 2001

J. Lladós, E. Martí, and J. J. Villanueva. Symbol Recognition by Error-Tolerant Subgraph Matching between Region Adjacency Graphs, IEEE TPAMI, vol. 23, no. 10, 2001.

GM in Document Analysis: Example 2

Approximate graph edit distance computation

- Exponential space and time complexity of graph edit distance.
- Cost matrix with substitution costs, deletion cost and insertion cost.
- Assignment problem.
- Munkres algorithm or Hungarian algorithm.

$$\mathbf{C} = \left[\begin{array}{cccc|cccc}
c_{1,1} & c_{1,2} & \cdots & c_{1,m} & c_{1,\varepsilon} & \infty & \cdots & \infty \\
c_{2,1} & c_{2,2} & \cdots & c_{2,m} & \infty & c_{2,\varepsilon} & \ddots & \vdots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \infty \\
c_{n,1} & c_{n,2} & \cdots & c_{n,m} & \infty & \cdots & \infty & c_{n,\varepsilon} \\
\hline
c_{\varepsilon,1} & \infty & \cdots & \infty & 0 & 0 & \cdots & 0 \\
\infty & c_{\varepsilon,2} & \ddots & \vdots & 0 & 0 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \infty & \vdots & \ddots & \ddots & 0 \\
\infty & \cdots & \infty & c_{\varepsilon,m} & 0 & \cdots & 0 & 0
\end{array}\right]$$

Figure credit: Riesen and Bunke IVC 2009

K. Riesen and H. Bunke. Approximate graph edit distance computation by means of bipartite graph matching. IVC, vol. 27, 2009.

GM in Document Analysis: Example 3

Integer linear programming for subgraph isomorphism

- Formulation of QAP as integer linear programming.
- Set of constraints that satisfy GM constraints.
- Integer solution with ILP.
- Still NP-hard but manageable with smaller graphs.

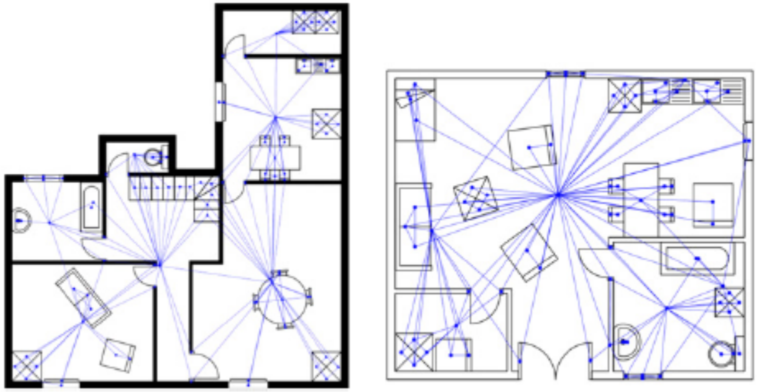
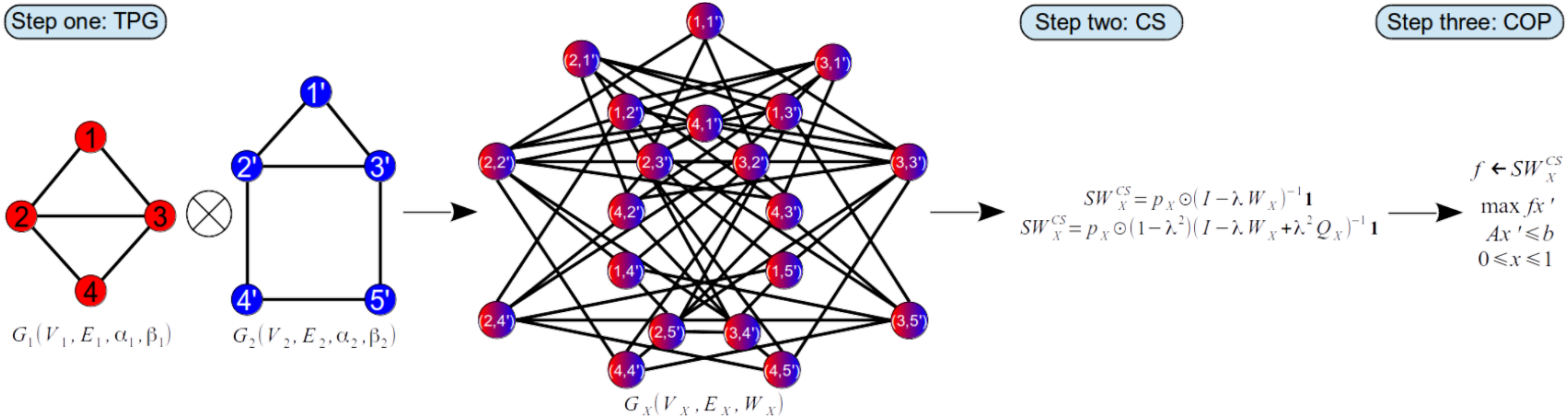


Figure credit: Le Bodic et al PR 2012

P. Le Bodic, P. Héroux, S. Adam, and Y. Lecourtier. An integer linear program for substitution-tolerant subgraph isomorphism and its use for symbol spotting in technical drawings, PR, vol. 45, no. 12, 2012.

GM in Document Analysis: Example 4

Higher order contextual similarities for subgraph isomorphism



A. Dutta, J. Lladós, H. Bunke, and U. Pal. Product Graph-based Higher Order Contextual Similarities for Inexact Subgraph Matching. ArXiv, 2017.

Summary: Graph Matching

- What is graph matching?
- Advances in graph matching?
- Graph Edit Distance
- Some examples employing graph matching, from PR and DIA works

Graph Embedding (GEM)

Evolution to Graph EMbedding (GEM)

- Graph matching and graph isomorphism

  [Messmer, 1995] [Sonbaty and Ismail, 1998]

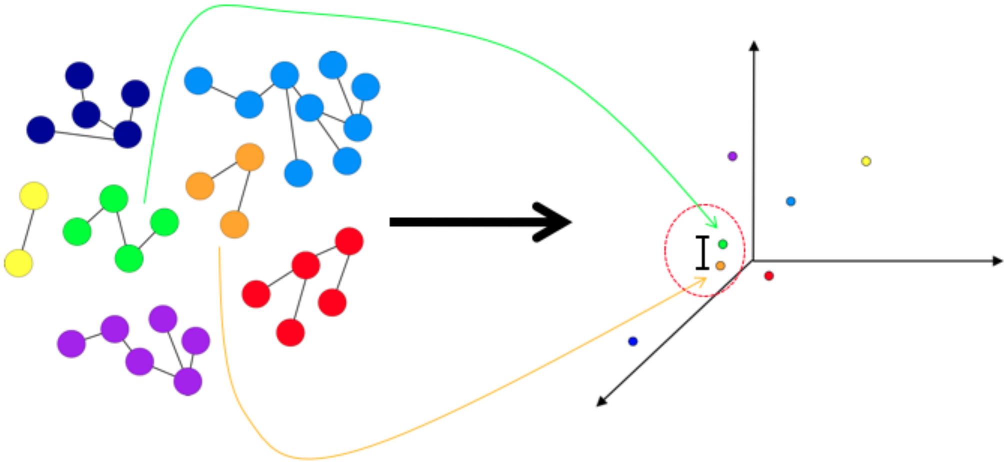
- Graph Edit Distance (GED)

  [Bunke and Shearer, 1998] [Neuhaus and Bunke, 2006]

- Graph EMbedding (GEM)

  [Luqman et al., 2009] [Sidere et al., 2009] [Gibert et al., 2011]

What is Graph EMbedding?



Luqman, M. M. (2012). Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. Ph.D. thesis. University of Tours, France and Autonoma University of Barcelona, Spain.

What is Graph EMbedding?

- Graph embedding is a methodology aimed at representing a whole graph, along with the attributes attached to its nodes and edges, as a point in a suitable vector space.

- By mapping a high dimensional graph into a point in suitable vector space, graph embedding permits to perform the basic mathematical computations which are required by various statistical pattern recognition techniques, and offers interesting solutions to the problems of graph clustering and classification.

Luqman, M. M. (2012). Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. Ph.D. thesis. University of Tours, France and Autonoma University of Barcelona, Spain.

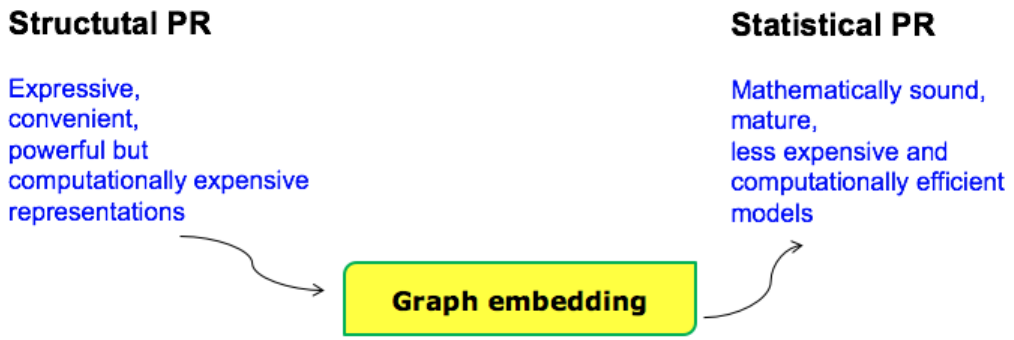
Why Graph Embedding (GEM) is needed?

- Graph have a powerful representations for extracting structural, topological and geometrical information of underlying content but lack in computational tools.

- GEM was a natural solution to enable graph-based representations to access computational efficient statistical models.

Luqman, M. M. (2012). Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. Ph.D. thesis. University of Tours, France and Autonoma University of Barcelona, Spain.

Graph Embedding (GEM)



Luqman, M. M. (2012). Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. Ph.D. thesis. University of Tours, France and Autonoma University of Barcelona, Spain.

Explicit GEM vs Implicit GEM

Explicit GEM

- embeds each input graph into a numeric feature vector
- provides more useful methods of GEM for PR
- can be employed in a standard dot product for defining an implicit graph embedding function

Implicit GEM

- computes scalar product of two graphs in an implicitly existing vector space, by using graph kernels
- does not permit all the operations that could be defined on vector spaces

Luqman, M. M. (2012). Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. Ph.D. thesis. University of Tours, France and Autonoma University of Barcelona, Spain.

Explicit GEM

- Graph probing based methods
- Spectral based graph embedding
- Dissimilarity based graph embedding

Luqman, M. M. (2012). Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. Ph.D. thesis. University of Tours, France and Autonoma University of Barcelona, Spain.

Explicit GEM

Graph probing based methods

[Wiener, 1947] [Papadopoulos et al., 1999] [Gibert et al., 2011] [Sidere et al., 2012]

number of nodes = 6
number of edges = 5
etc.

v = 6,5, …

Explicit GEM

Spectral based methods

[Harchaoui, 2007] [Luo et al., 2003] [Robleskelly and Hancock, 2007]

	1				
1		1			
	1		1		
		1		1	1
			1		
			1		

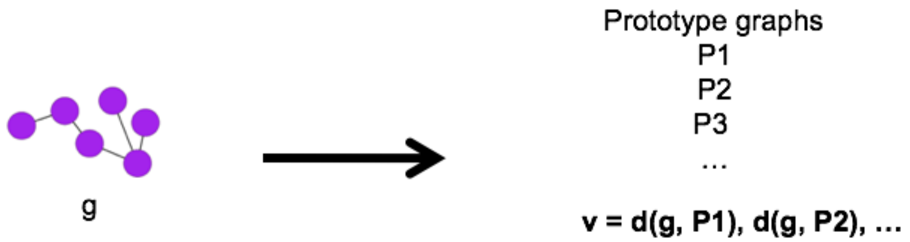
Spectral graph theory employing the adjacency and Laplacien matrices

- Eigen values and Eigen vectors
- PCA, ICA, MDS

Explicit GEM

Dissimilarity based methods

[Pekalska et al., 2005] [Ferrer et al., 2008] [Riesen, 2010] [Bunke et al., 2011]



Explicit GEM

Graph feature extraction based methods

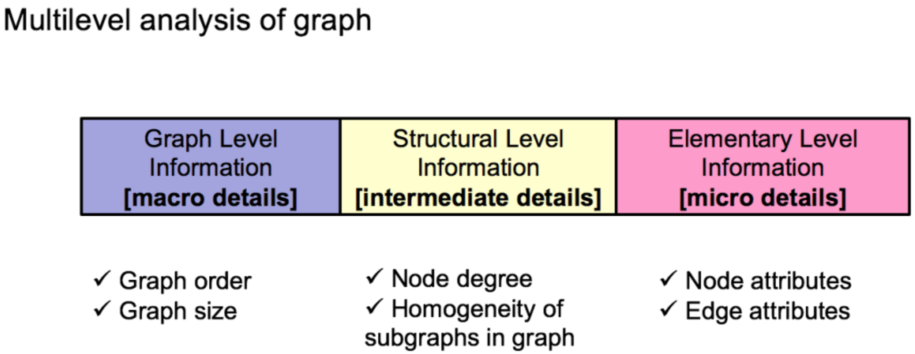
- Node information
- Edge information
- Structure
- Topology
- Geometry

Muhammad Muzzamil Luqman, Jean-Yves Ramel, Josep Lladós, Thierry Brouard: Fuzzy multilevel graph embedding. Pattern Recognition 46(2): 551-565 (2013)

Nicholas Dahma, Horst Bunke, Terry Caelli, Yongsheng Gao. Efficient subgraph matching using topological node feature constraints, Pattern Recognition 48 (2015) 317330.

Explicit GEM

Graph feature extraction based methods - FMGE

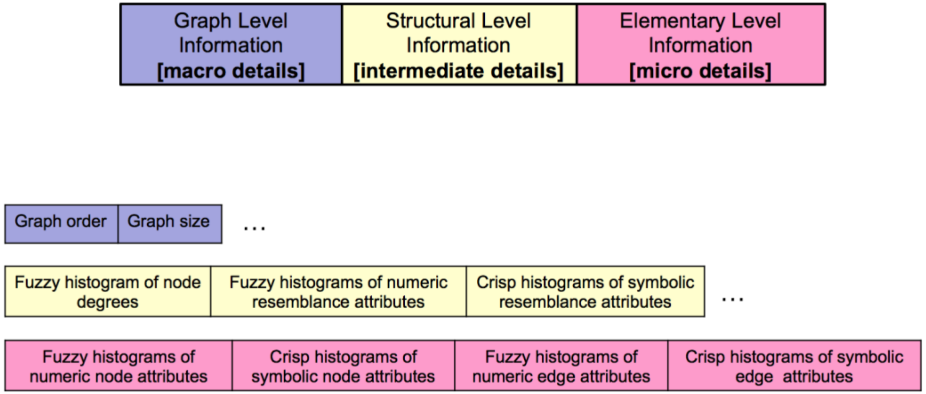


Muhammad Muzzamil Luqman, Jean-Yves Ramel, Josep Lladós, Thierry Brouard: Fuzzy multilevel graph embedding. Pattern Recognition 46(2): 551-565 (2013)

Explicit GEM

Graph feature extraction based methods - FMGE

- Numeric feature vector embeds a graph, encoding Numeric information by fuzzy histograms and Symbolic information by crisp histograms



Explicit GEM

Graph feature extraction based methods - FMGE

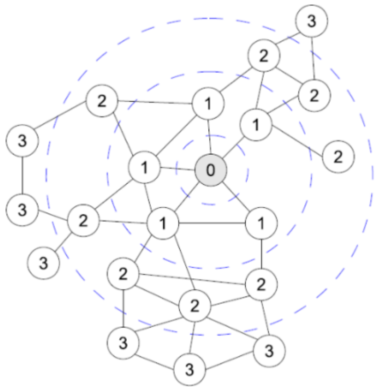
- Equal-size numeric feature vectors for each input graphs

Graph order	Graph size	Embedding of node degree	Embedding(s) of subgraph(s) homogenity	Embedding(s) of node attribute(s)	Embedding(s) of edge attribute(s)

Muhammad Muzzamil Luqman, Jean-Yves Ramel, Josep Lladós, Thierry Brouard: Fuzzy multilevel graph embedding. Pattern Recognition 46(2): 551-565 (2013)

Explicit GEM

Graph feature extraction based methods - Improved FMGE



Morgan index for encoding Topological n-neighbourhood feature

Hana Jarraya, Muhammad Muzzamil Luqman, Jean-Yves Ramel: Improving Fuzzy Multilevel Graph Embedding Technique by Employing Topological Node Features: An Application to Graphics Recognition. GREC 2015: 117-132

Explicit GEM

Topological Embedding

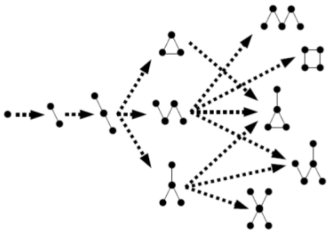
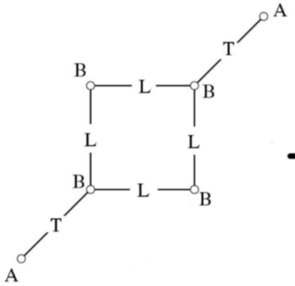


Fig. 1. The non-isomorphic graph network used to embed the topology.



(a)

	topology	6	6	8	8	0	2
vertices	A	2	2	4	4	0	2
	B	4	10	20	28	0	6
edges	T	0	2	4	4	0	2
	L	0	4	12	20	0	4

(b)

Fig. 2. Matrix (b) corresponding to vectorial signature of graph presented in (a).

Sidere,N.,Héroux,P.,Ramel,J.Y.:Vector representation of graphs:Application to the classification of symbols and letters. In: ICDAR. pp. 681–685. IEEE Computer Society (2009)

Explicit GEM

- Attribute Statistics based Embedding

Simple and efficient way of expressing the labelling information stored in nodes and edges of graphs in a rather naive feature vector.

Frequencies of appearances of very simple subgraph structures such as nodes with certain labels or node-edge-node structures with specific label sequences.

Gibert, J., Valveny, E., Bunke, H.: Graph embedding in vector spaces by node attribute statistics. Pattern Recognition 45(9), 3072–3083 (2012)

Explicit GEM

- Constant shift embedding

  [Jouili, S., Tabbone, S.: Graph embedding using constant shift embedding. In: Proceedings of the 20th International conference on Recognizing patterns in signals, speech, images, and videos. pp. 83–92. ICPR’10]

Some applications of Explicit GEM from literature

- Graph similarity
- Graph classification
- Graph clustering
- Symbol recognition/classification/clustering
- Chemical molecules recognition/classification/clustering
- Fingerprint recognition

Some applications of Explicit GEM from literature

- Subgraph spotting
- Symbol spotting/retrieval
- Comics retrieval
- QBE in document images
- Focused retrieval in document images

Explicit GEM

Limitations:

- Not many methods for both directed and undirected attributed graphs
- No method explicitly addresses noise sensitivity of graphs
- Expensive deployment to other application domains
- Time complexity
- Loss of topological information
- Loss of matching between nodes
- No graph embedding based solution to answer high level semantic problems for graphs

Implicit GEM (Graph kernels)

What is implicit GEM?

Methods based on graph kernels.

Graph kernel is a function that can be thought of as a dot product in some implicitly existing vector space. Instead of mapping graphs from graph space to vector space and then computing their dot product, the value of the kernel function is evaluated in graph space.

Conte, D., Ramel, J. Y., Sidère, N., Luqman, M. M., Gaüzère, B., Gibert, J., … Vento, M. (2013). A comparison of explicit and implicit graph embedding methods for pattern recognition. 9th IAPR-TC15 Workshop on Graph-Based Representations in Pattern Recognition (GbR2013), 7877 LNCS, 81–90.

Bunke, H., Riesen, K.: Recent advances in graph-based pattern recognition with applications in document analysis. Pattern Recognition 44(5), 1057–1067 (2011)

Implicit GEM (Graph kernels)

What is implicit GEM?

- Graph kernels can be intuitively understood as functions measuring the similarity of pairs of graphs.
- They allow kernelized learning algorithms such as support vector machines to work directly on graphs, without having to do feature extraction to transform them to fixed-length, real-valued feature vectors.

Some graph kernels (implicit GEM methods)

- Laplacian Graph Kernel

- Treelet Kernel
  - A graph kernel based on a bag of non linear patterns which computes an explicit distribution of each pattern within a graph.
  - This method explicitly enumerates the set of treelets included within a graph. The set of treelets, denoted T , is defined as the 14 trees having a size lower than or equals to 6 nodes.
  - This vector representation may be of very high dimension since it may encode all possible treelets according to all possible nodes and edges labellings defined for a graph family.

Gauzère, B., Brun, L., Villemin, D.: Two new graphs kernels in chemoinformatics. Pattern Recognition Letters 33(15), 2038 – 2047 (2012)

Some graph kernels (implicit GEM methods)

Random Walk Kernel

- Conceptually performs random walks on two graphs simultaneously, then counts the number of paths that were produced by both walks.
- This is equivalent to doing random walks on the direct product of the pair of graphs, and from this, a kernel can be derived that can be efficiently computed
- Walks are sequences of nodes that allow repetitions of nodes

Michel Neuhaus, Horst Bunke: A Random Walk Kernel Derived from Graph Edit Distance. SSPR/SPR 2006: 191-199

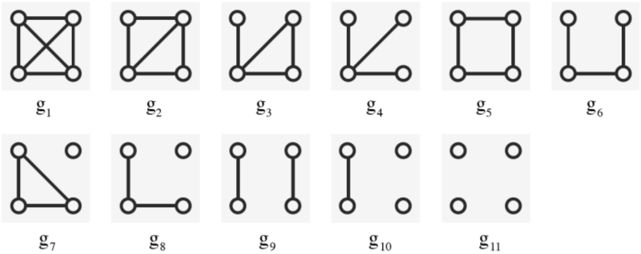
Some graph kernels (implicit GEM methods)

Graphlet Kernel

- Graphlets := graphs of size {3, 4, 5}.
- $\mathfrak{G} = \{g_{1_k}, g_{2_k}, \ldots, g_{N_k}\}$ be the set of size k graphlets and G be a graph of size n.
- Let f_G be a vector of length N_k such that $f_{G_i} = \#((g_i) \in G)$
- Given two graphs G, G' of size $n \geq k$, the graphlet kernel K_g is defined as

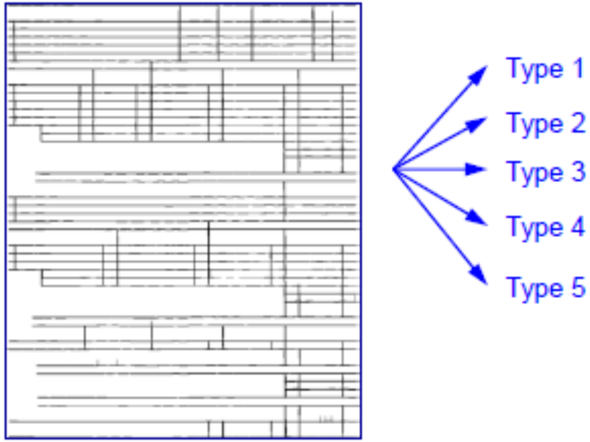
$$K_g(G, G') = f_g^T . f_g'$$

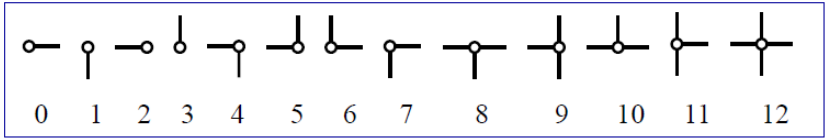
- Size 4 graphlets



N. Shervashidze, S. V. N. Vishwanathan, T. Petri, K. Mehlhorn and K. Borgwardt, "Efficient graphlet kernels for large graph comparison". AISTATS, 2009.

Graph Lattice Approach





13 Primitive Junction Types

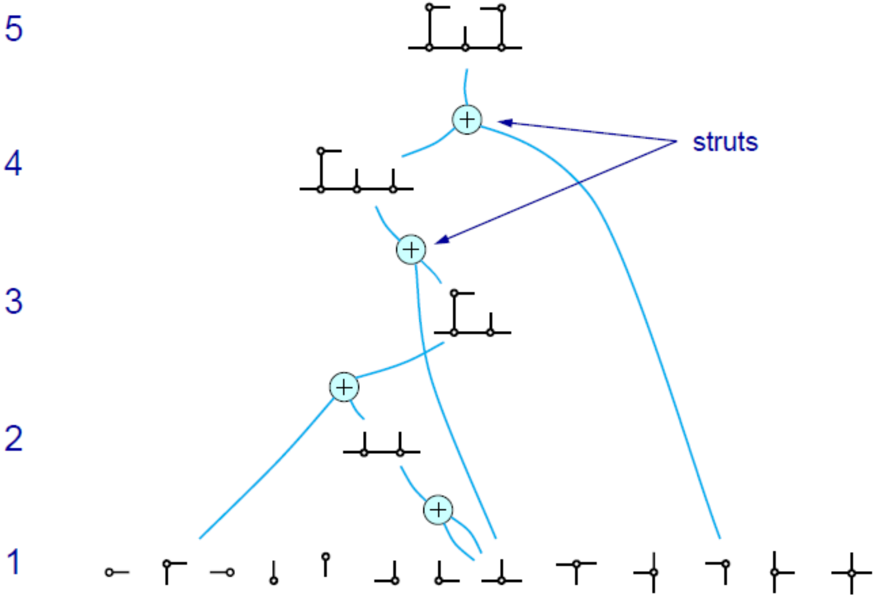
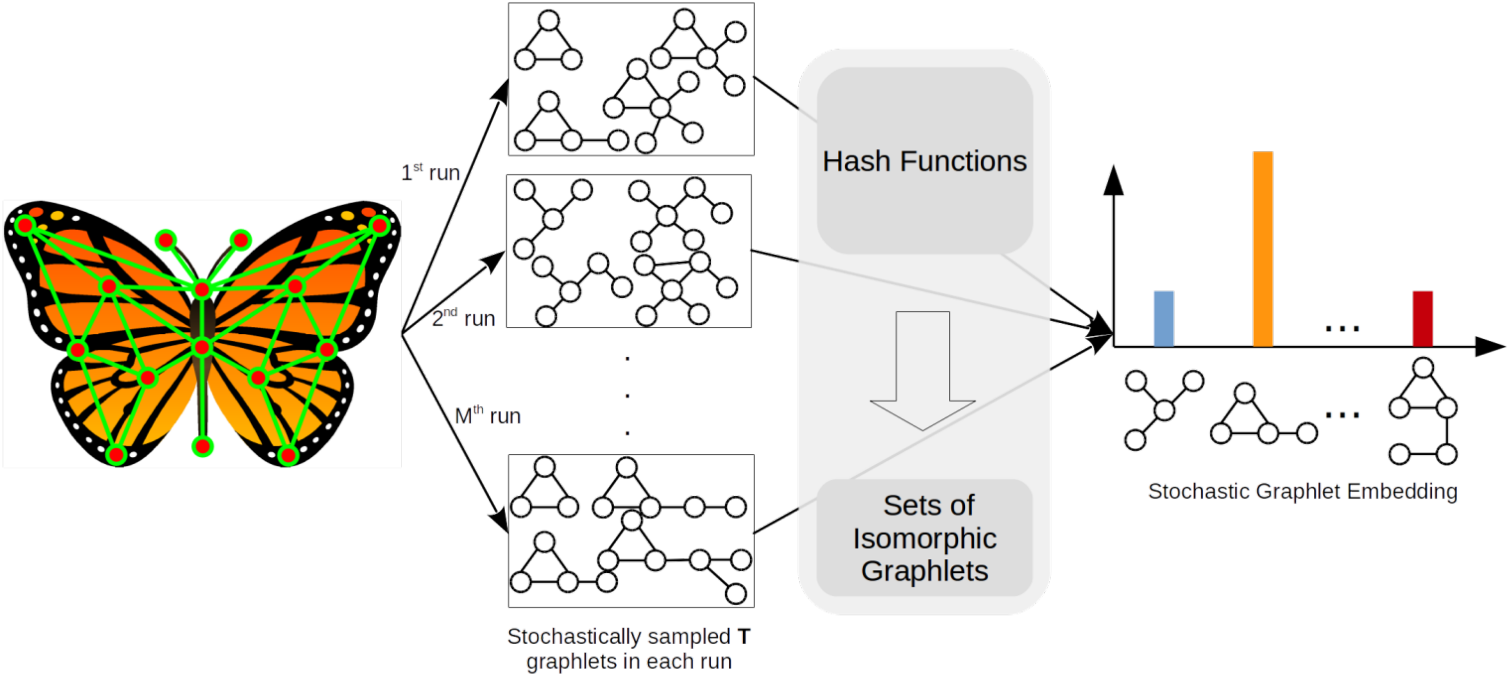


Figure credit: Eric Saund ICDAR 2011

E. Saund, “A graph lattice approach to maintaining and learning dense collections of subgraphs as image features”. IEEE TPAMI vol. 35, no. 10, pp. 2323–2339, 2013.

Implicit GEM

Stochastic graphlet embedding



A. Dutta, and H. Sahbi. High Order Stochastic Graphlet Embedding for Graph-Based Pattern Recognition. ArXiv, 2017.

Implicit GEM (Graph kernels)

Properties and Limitations

An implicit graph embedding satisfies all properties of a dot product.

Since it does not explicitly map a graph to a point in vector space, a strict limitation of implicit graph embedding is that it does not permit all operations that could be defined on vector spaces.

Conte, D., Ramel, J. Y., Sidère, N., Luqman, M. M., Gaüzère, B., Gibert, J., … Vento, M. (2013). A comparison of explicit and implicit graph embedding methods for pattern recognition. 9th IAPR-TC15 Workshop on Graph-Based Representations in Pattern Recognition (GbR2013), 7877 LNCS, 81–90.

Bunke, H., Riesen, K.: Recent advances in graph-based pattern recognition with applications in document analysis. Pattern Recognition 44(5), 1057–1067 (2011)

Summary: Graph Embedding

- Evolution to GEM?
- What is Graph Embedding?
- Explicit GEM
    - Some methods of Explicit GEM
    - Some applications in literature
- Implicit GEM or Graph kernels
    - Some graph kernels

Coffee break

10h30 – 11h00

Session-2 (11h - 12h30)

1. Graph indexing, graph retrieval, subgraph spotting and diffusion, serialization
2. Neural network on graphs
3. Programming languages, evaluation protocols, datasets and Programming

   Hands-on: Graph classification with RW kernel
4. Discussion (12h15 – 12h30)

Graph Indexing, Graph Retrieval, Subgraph Spotting
and
Graph Diffusion, Graph Serialization

Graph indexing, retrieval and subgraph spotting

What is subgraph spotting?

- The research problem of searching a query graph in a database of graphs is termed as “subgraph spotting”.
- This means that for a given query attributed graph the goal is to retrieve every graph in the database which contains this query graph and to provide node correspondences between the query and each of the result graphs.

Graph indexing, retrieval and subgraph spotting

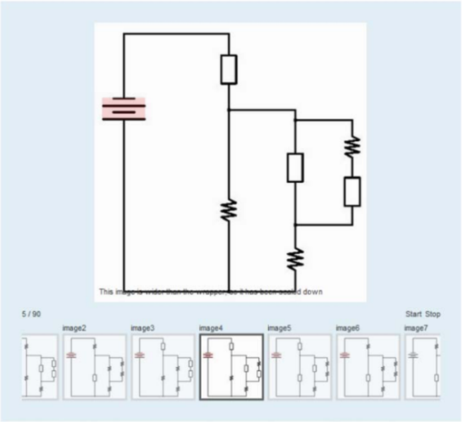
How it is different from subgraph matching?

- Subgraph matching generally refers to matching two graphs, where size of one graph is greater than (or equal to) the other

- Subgraph spotting generally refers to the problem when we have to find a graph in a database of graphs of larger size

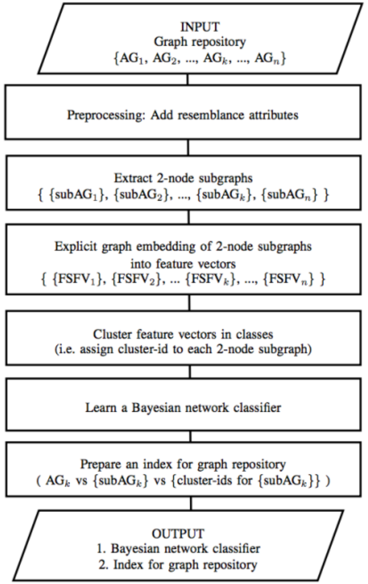
Subgraph Spotting through Explicit Graph Embedding: An Application to Content Spotting in Graphic Document Images

(a) Query image.



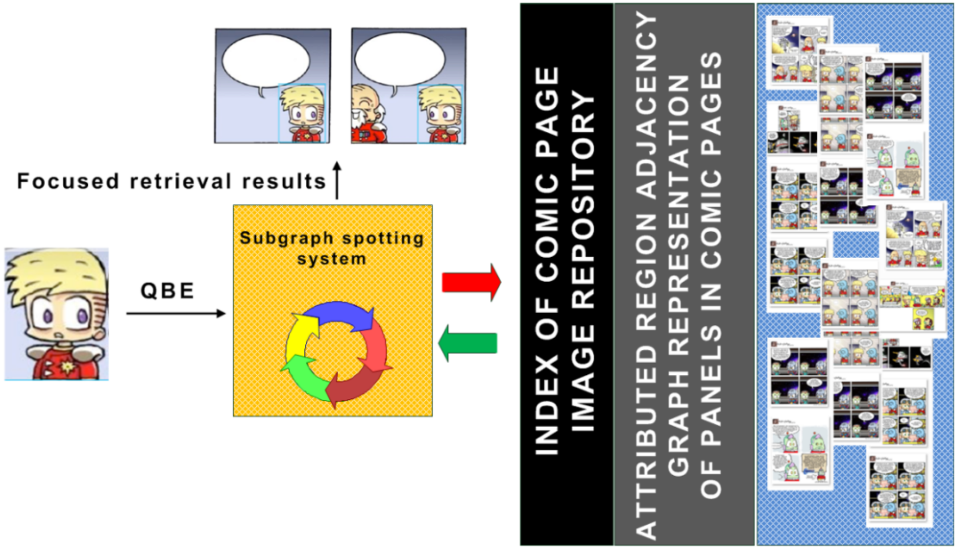
Luqman, M. M., Ramel, J. Y., Lladós, J., & Brouard, T. (2011). Subgraph spotting through explicit graph embedding: An application to content spotting in graphic document images. International Conference on Document Analysis and Recognition, ICDAR, 870–874.

Subgraph Spotting through Explicit Graph Embedding: An Application to Content Spotting in Graphic Document Images



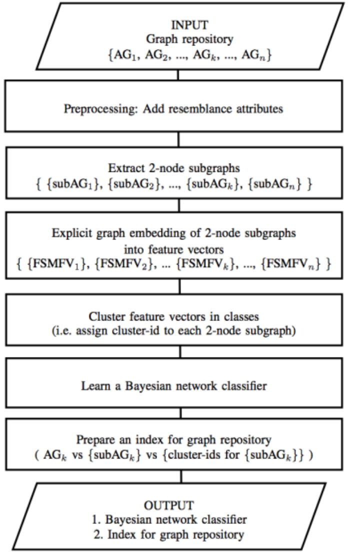
Luqman, M. M., Ramel, J. Y., Lladós, J., & Brouard, T. (2011). Subgraph spotting through explicit graph embedding: An application to content spotting in graphic document images. International Conference on Document Analysis and Recognition, ICDAR, 870–874.

Automatic indexing of comic page images for query by example based focused content retrieval



Luqman, M. M., Ho, H. N., Burie, J., & Ogier, J. (2013). Automatic indexing of comic page images for query by example based focused content retrieval. In Tenth IAPR International Workshop on Graphics RECognition (GREC) (pp. 153–157).

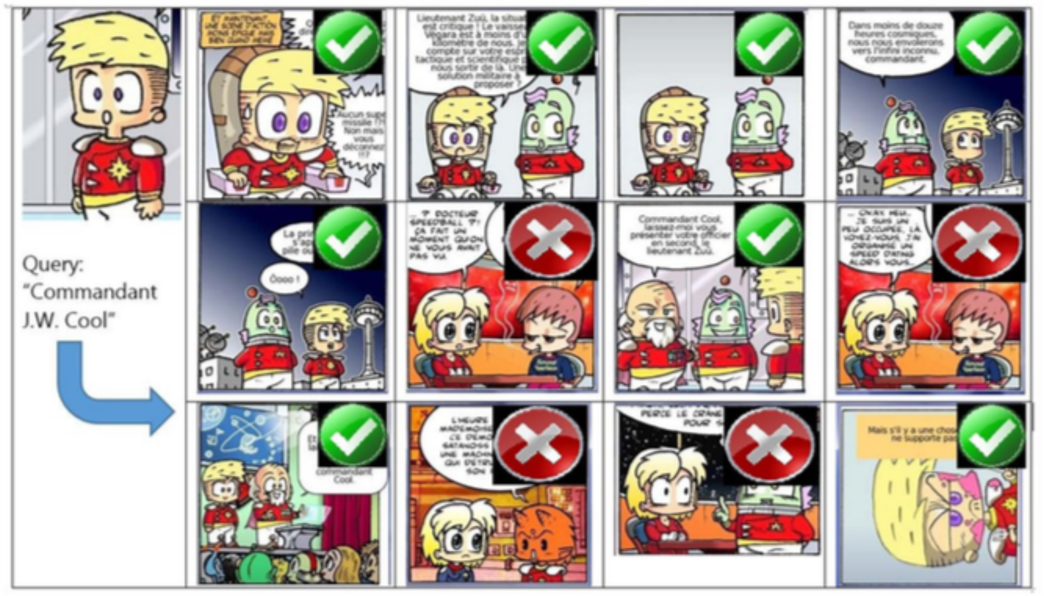
Automatic indexing of comic page images for query by example based focused content retrieval



Automatic indexation of a graph repository.

Luqman, M. M., Ho, H. N., Burie, J., & Ogier, J. (2013). Automatic indexing of comic page images for query by example based focused content retrieval. In Tenth IAPR International Workshop on Graphics RECognition (GREC) (pp. 153–157).

Content-based Comic Retrieval Using Multilayer Graph Representation and Frequent Graph Mining



Le, T., Luqman, M. M., Burie, J., & Ogier, J. (2015). Content-based Comic Retrieval Using Multilayer Graph Representation and Frequent Graph Mining. 13th International Confrence on Document Analysis and Recognition - ICDAR'15, 15–19.

Content-based Comic Retrieval Using Multilayer Graph Representation and Frequent Graph Mining

- An adaptation of bag-of-words model to graph domain
- Extract Frequent Patterns from the database graphs and construct an index
- Extract frequent patterns from query graph and match with the index
- The intersection of all the frequent patterns of query graph gives list of result graphs
- The node list of a result graph gives the spotted subgraph

Le, T., Luqman, M. M., Burie, J., & Ogier, J. (2015). Content-based Comic Retrieval Using Multilayer Graph Representation and Frequent Graph Mining. 13th International Confrence on Document Analysis and Recognition - ICDAR’15, 15–19.

Fuzzy generalized median graphs computation: Application to content-based document retrieval

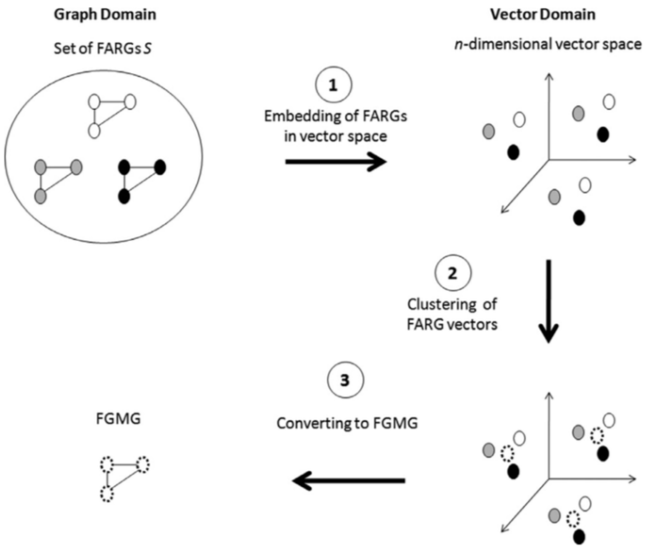


Fig. 1. The three main steps of the proposed FGMG computation algorithm.

Ramzi Chaieb, Karim Kalti, Muhammad Muzzamil Luqman, Mickaël Coustaty, Jean-Marc Ogier, Najoua Essoukri Ben Amara: Fuzzy generalized median graphs computation: Application to content-based document retrieval. Pattern Recognition 72: 266-284 (2017)

Fuzzy generalized median graphs computation: Application to content-based document retrieval

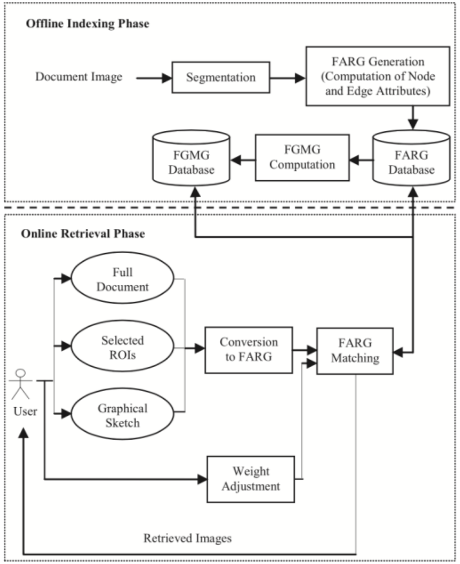


Fig. 6. Block diagram of the proposed CBDR approach using FGMGs.

Ramzi Chaieb, Karim Kalti, Muhammad Muzzamil Luqman, Mickaël Coustaty, Jean-Marc Ogier, Najoua Essoukri Ben Amara: Fuzzy generalized median graphs computation: Application to content-based document retrieval. Pattern Recognition 72: 266-284 (2017)

Graph Diffusion

- Spreading or movement of information between nodes along a graph's edges is called *graph diffusion*.
- Reversible Markov process. $Q^{(t+1)} = EQ^{(t)}E^T + I$
- Application on
  - affinity learning for object retrieval.
  - improving retrieval quality in multiwriter scenario.

1. X. Yang, L. Prasad and L. J. Latecki. Affinity Learning with Diffusion on Tensor Product Graph. IEEE TPAMI, vol. 35, no. 1, 2012.
2. P. Riba, A. Dutta, S. Dey, J. Lladós and A. Fornés. Improving Information Retrieval in Multiwriter Scenario by Exploiting the Similarity Graph of Document Terms. To be presented in ICDAR, 2017

Diffusion on Tensor Product Graph

- Pairwise similarity is unreliable and sensitive to noise.
- Diffused similarity in the context of other data points are better reliable.
- Tensor product graph takes into account higher order information.
- Diffusion on TPG is equivalent to an iterative process on the original graph.

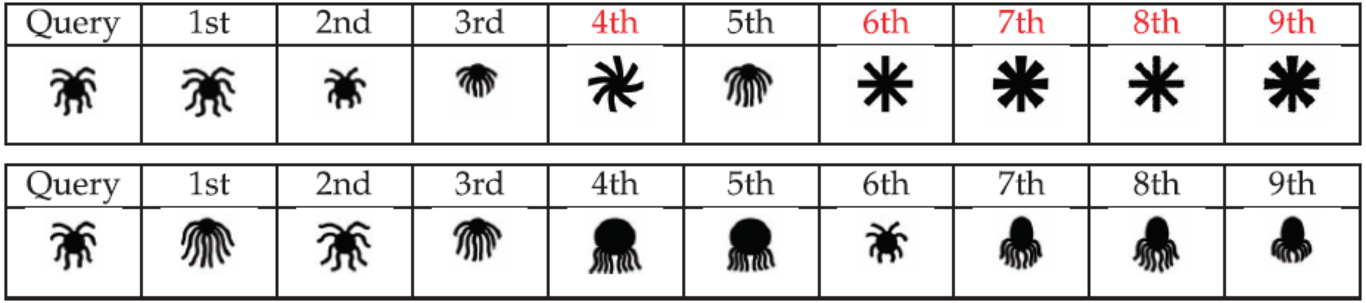
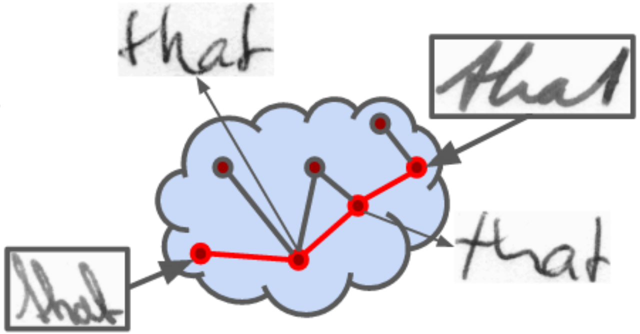


Figure credit: Yang et al TPAMI 2012

X. Yang, L. Prasad and L. J. Latecki. Affinity Learning with Diffusion on Tensor Product Graph. IEEE TPAMI, vol. 35, no. 1, 2012.

Information retrieval in multiwriter scenario

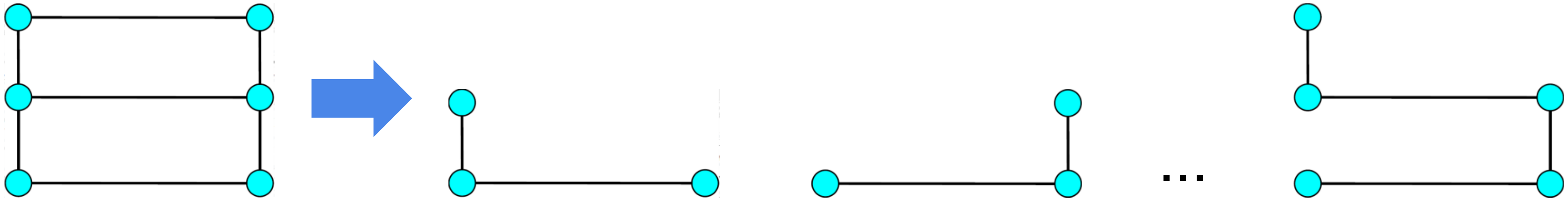
- Graph with each node as a document term and similarity between them as edge weight.
- Different graph analytics: *diffusion*, *shortest path* to get a different similarity value.
- Improved performance in multiwriter scenario.
- Information retrieval using multiple queries.



P. Riba, A. Dutta, S. Dey, J. Lladós and A. Fornés. Improving Information Retrieval in Multiwriter Scenario by Exploiting the Similarity Graph of Document Terms. To be presented in ICDAR, 2017 (Presentation on 14th Nov.)

Graph serialization

- One dimensional structure. Graph paths.

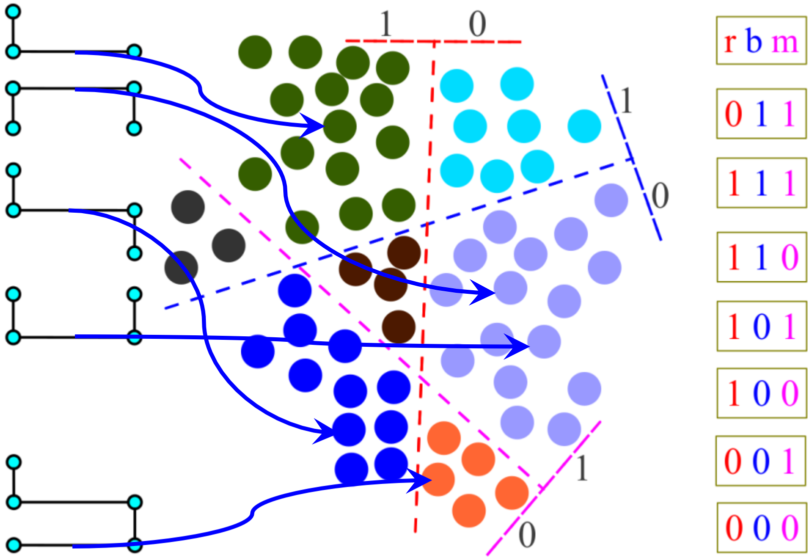


- Shape descriptors. Ex: Zernike moments, Hu moments etc.
- Indexing of graph paths.

A. Dutta, J. Lladós and U. Pal. A symbol spotting approach in graphical documents by hashing serialized graphs. In PR, vol. 46, no. 3, pp. 752-768, 2013.

Graph serialization

- Hashing of serialized subgraphs.
- Locality sensitive hashing.
- Retrieval of paths and spatial voting for symbol spotting.



A. Dutta, J. Lladós, and U. Pal. A symbol spotting approach in graphical documents by hashing serialized graphs. PR, vol. 46, no. 3, pp. 752-768, 2013.
P. Indyk and R. Motwani. “Approximate nearest neighbors: towards removing the curse of dimensionality”. ACMSTOC, pp. 604-613, 1998.

Summary: Graph indexing, retrieval, subgraph spotting, diffusion, serialization

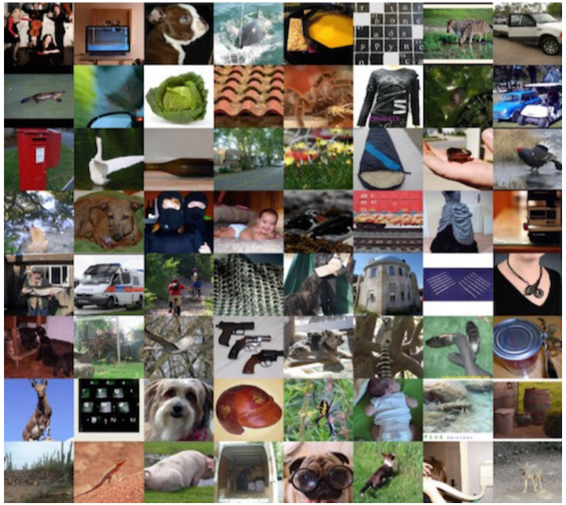
- What is graph indexing, retrieval and subgraph spotting?
  - Examples of systems from literature

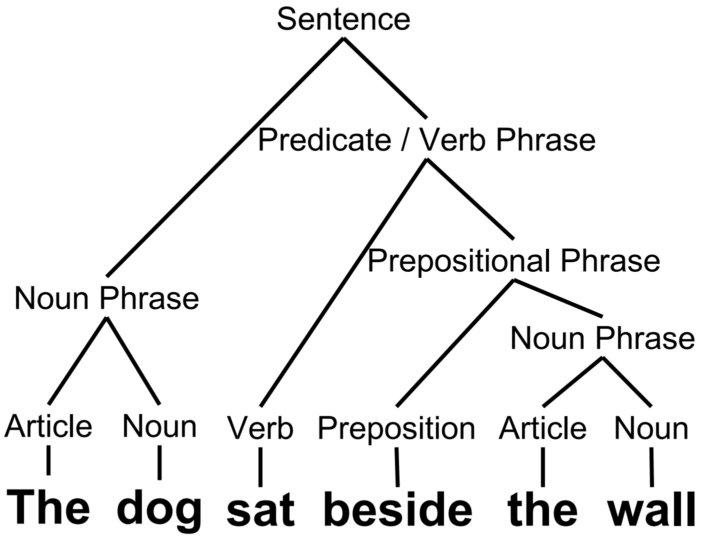
- What is graph diffusion and serialisation?
  - Examples of systems from literature

Neural network on graphs

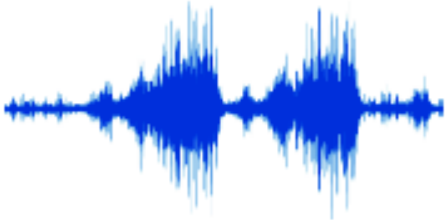
Success story of deep learning

IMAGENET





Natural Language Processing (NLP)



Speech Data

Slide credit: Kipf et al. Deep Learning on Graphs with Graph Convolutional Networks

Evolution of deep learning

Perceptron
Rosenblatt

Backprop
Werbos

First NIPS

SVM
Vapnik

CNN
LeCun

Autoencoder
LeCun, Hinton

IMAGENET

facebook
AI Research

DeepMind
AlphaGo

Microsoft
Speech Recognition

1958 1959 1982 1987 1995 1997 1998 1999 2006 2010 2012 2014 2015 2016

Visual cortex
Hubel & Wiesel

Neurocognitron
Fukushima

RNN / LSTM
Schmidhuber

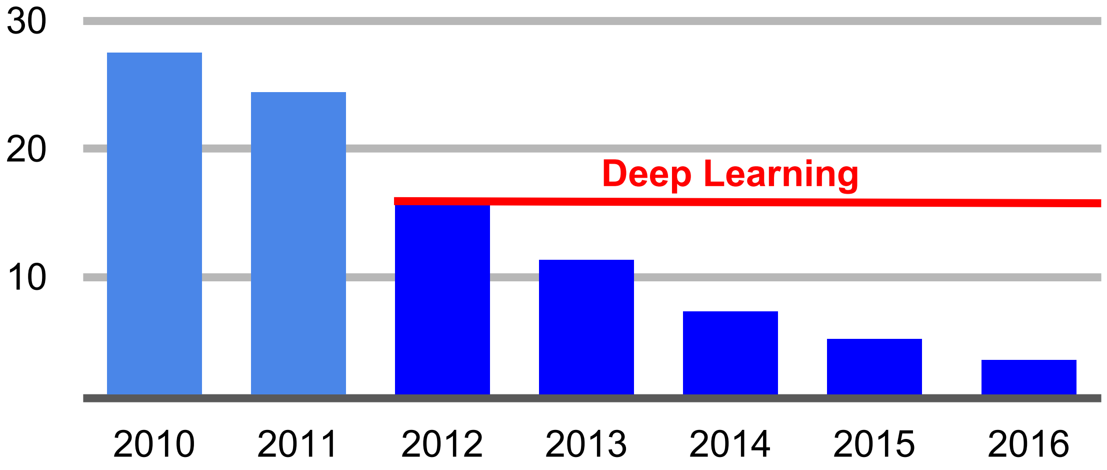
NVIDIA GEFORCE
First GPU

ImageNet
breakthrough
Krizhevsky

TensorFlow
OpenAI
TESLA
Autonomous
cars

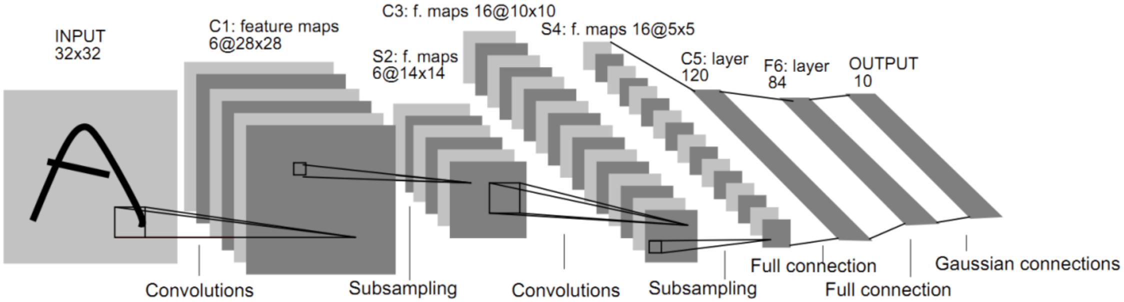
Slide credit: M. Bronstein et al. Geometrical Deep Learning, Tutorial, CVPR, 2017

Breakthrough in image recognition



Slide credit: M. Bronstein et al. Geometrical Deep Learning, Tutorial, CVPR, 2017

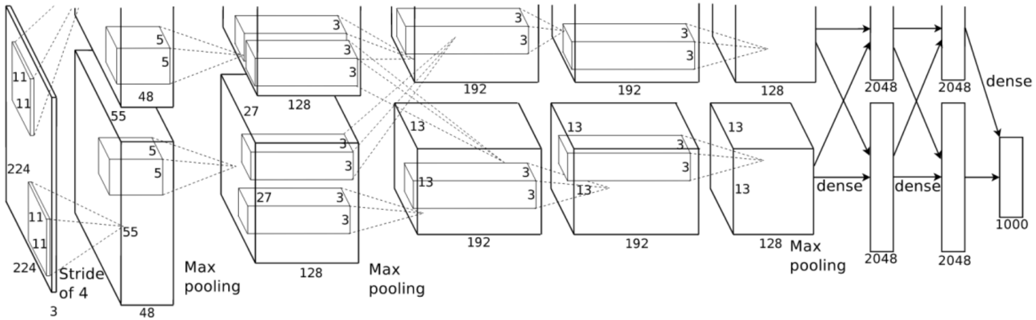
CNN: LeNet 5



- 3 convolutional + 1 fully connected layer
- 1M parameters
- Training set: MNIST 70K images
- Trained on CPU
- `tanh` non-linearity

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. IEEE, 1998.

CNN: AlexNet

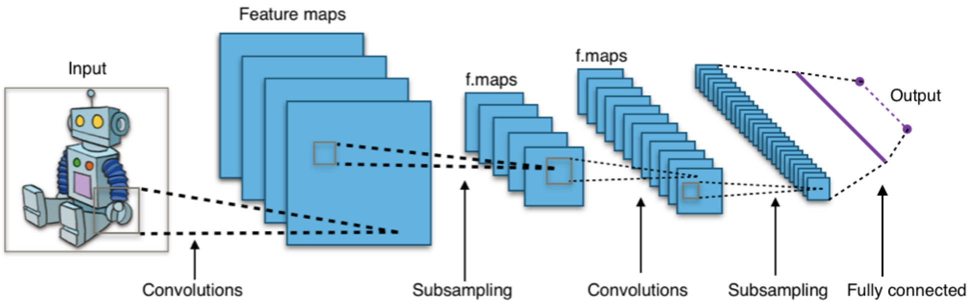


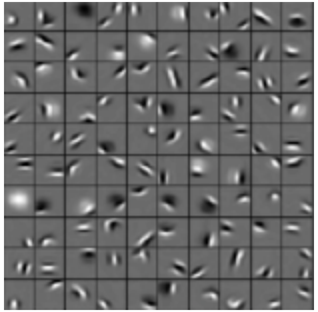
- 5 convolutional + 3 fully connected layer
- 60M parameters
- Trained on ImageNet 1.5M images
- Trained on GPU
- ReLU non-linearity
- Dropout regularization

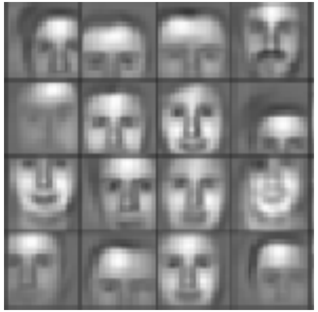
A. Krizhevsky, I. Sutskever and G. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. NIPS, 2012.

Convolutional neural network

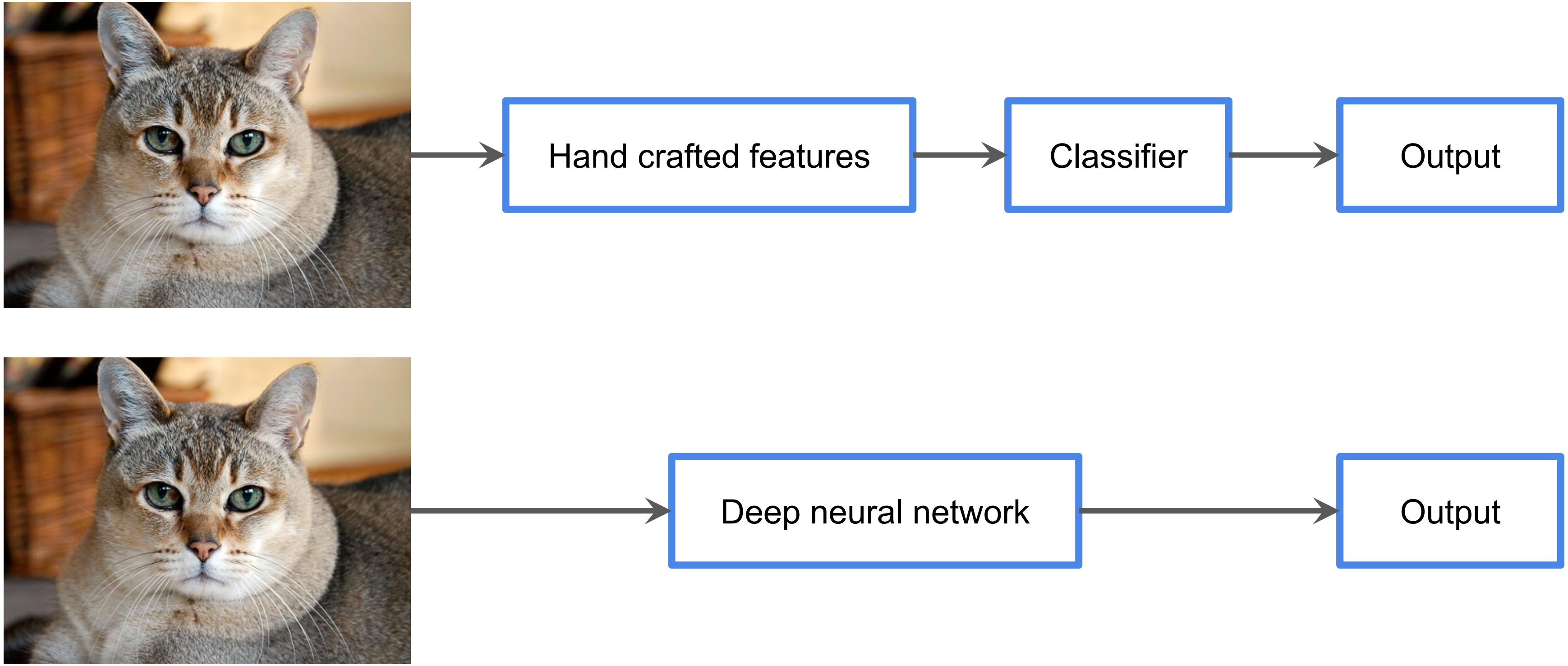
- Hierarchical compositionality
- Weight sharing
- Big data
- Computational power







Traditional vs “deep” learning



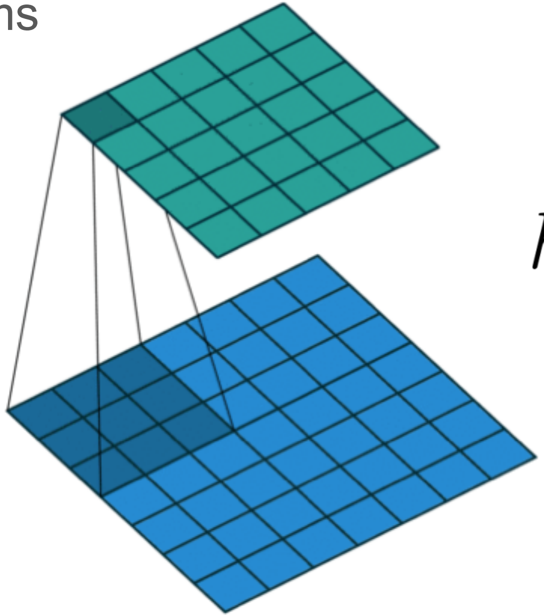
CNN: Message passing in a grid graph

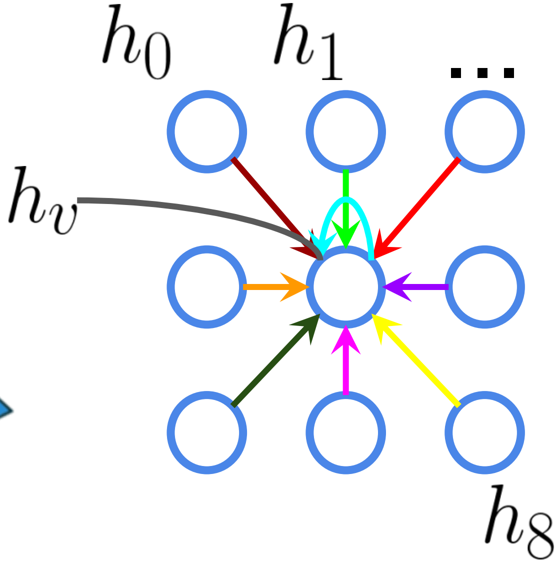
- Individual message transforms

$$W_v h_v$$

- Sum everything up

$$\sum_v W_v h_v$$





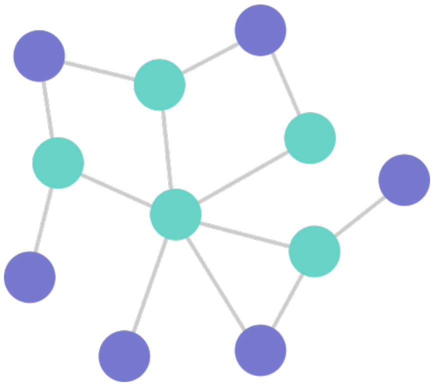
- Full update

$$h_v^{t+1} = \sigma(W_0^t h_0^t + W_1^t h_1^t + \ldots + W_v^t h_v^t + \ldots + W_8^t h_8^t)$$

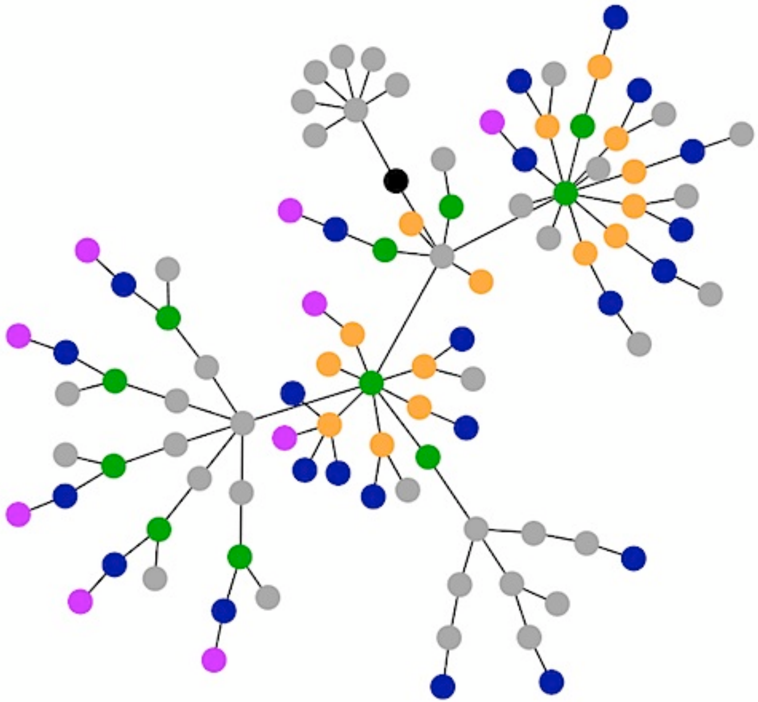
Animation by V. Dumoulin

Graph structured data

What if the data look like this?



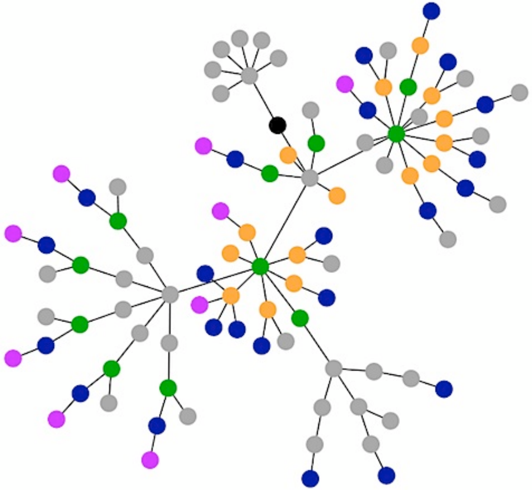
or this:



Graph structured data

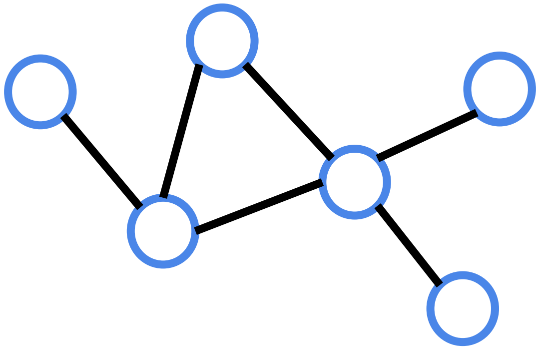
Real world examples:

- Social networks
- World wide web
- Protein interaction networks
- Telecommunication networks
- Knowledge graphs
- ...

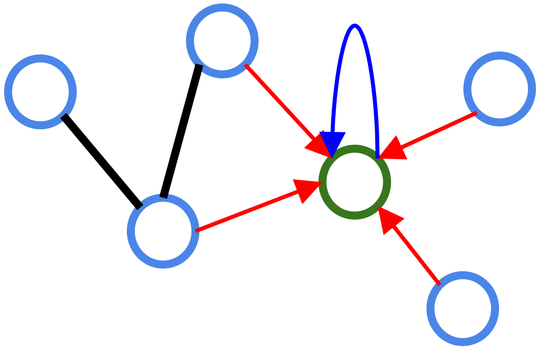


Message passing on graphs

Consider this undirected graph:



Calculate update for node in green:



Update rule:

$$h_v^{t+1} = \sigma\left(h_v^t W_0^t + \frac{1}{c_{vw}} \sum_{w \in \mathcal{N}_v} h_w^t W_1^t\right)$$

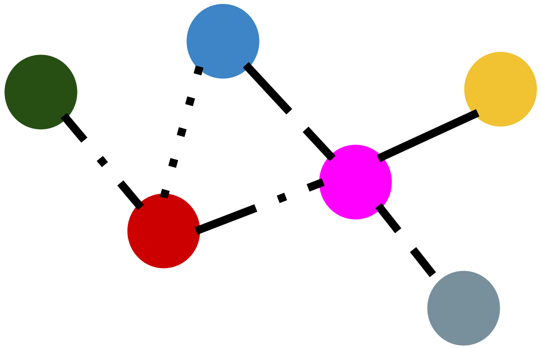
More general or simpler function also can be chosen

1. J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, G. E. Dahl. Neural Message Passing for Quantum Chemistry. ICML, 2017.
2. T. Kipf, M. Welling, Semi-Supervised Classification with Graph Convolutional Networks. ICLR, 2017

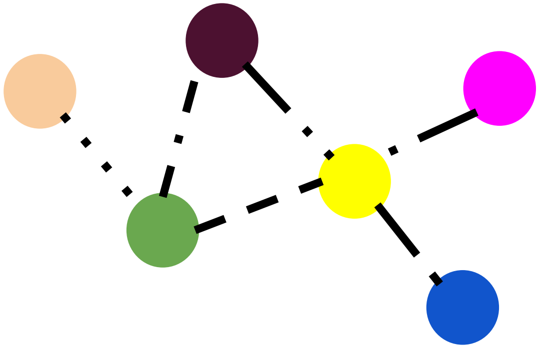
Several iteration of message passing

Node and edge updation:

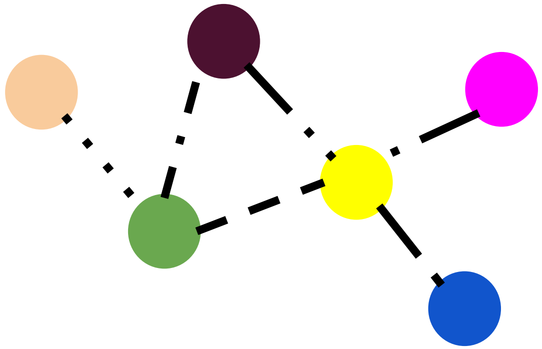
Initial stage:



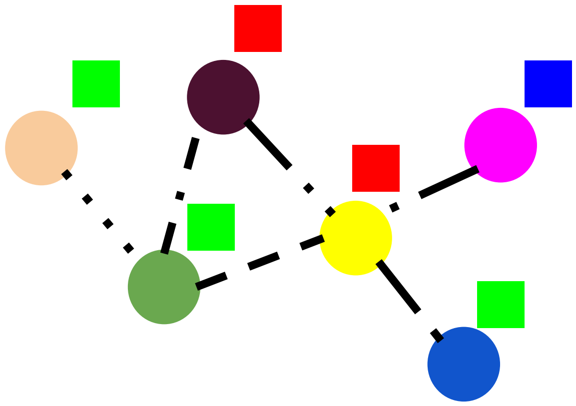
Final stage:



Graph wise classification



Node wise classification



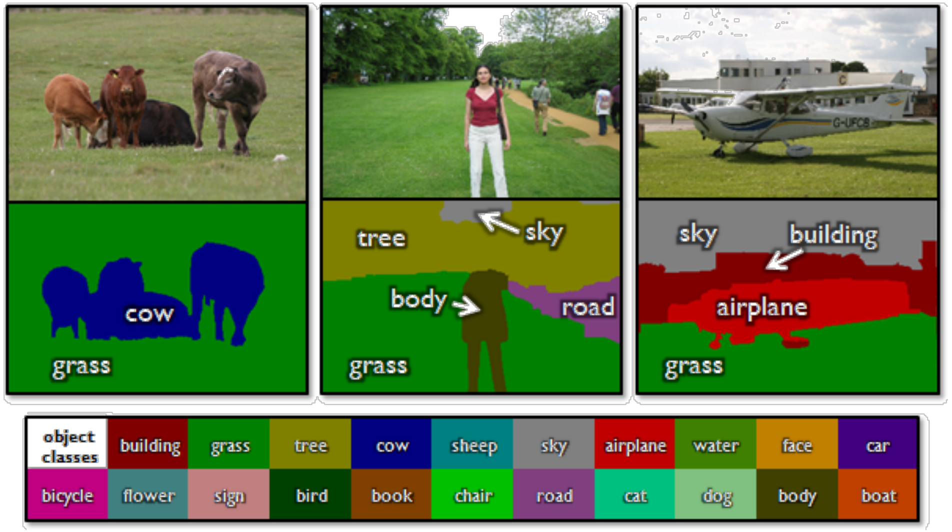


Figure credit: Shotton et al IJCV 2007

Neural Message Passing

Message function:

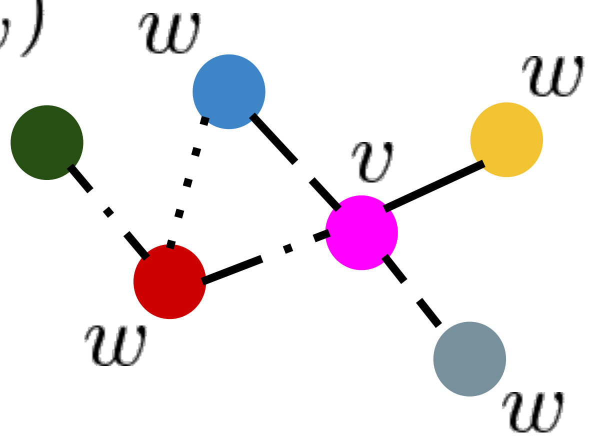
$$m_v^{t+1} = \sum_{w \in \mathcal{N}_v} M(h_v^t, h_w^t, e_{vw})$$

Update function:

$$h_v^{t+1} = U(h_v^t, m_v^{t+1})$$

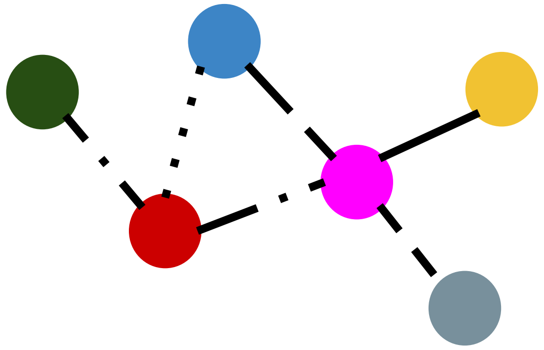
Readout function:

$$\hat{y} = R(\{h_v^T \mid v \in G\})$$



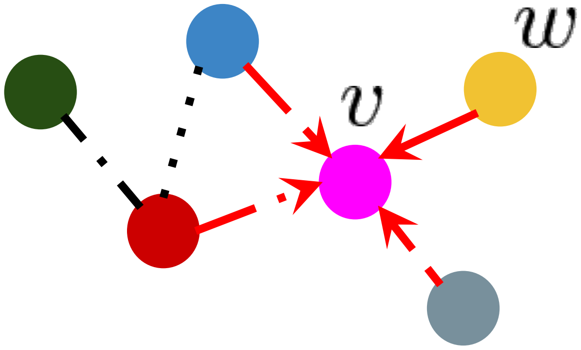
J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, G. E. Dahl. Neural Message Passing for Quantum Chemistry. ICML, 2017.

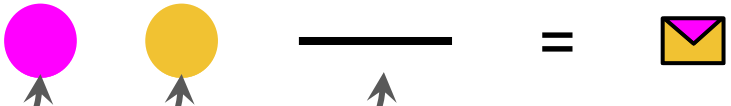
Running Example



Message passing

Message function:

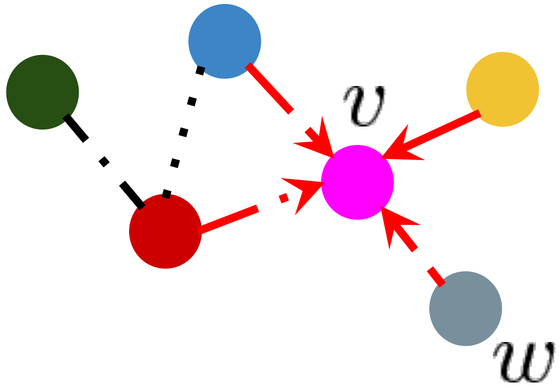


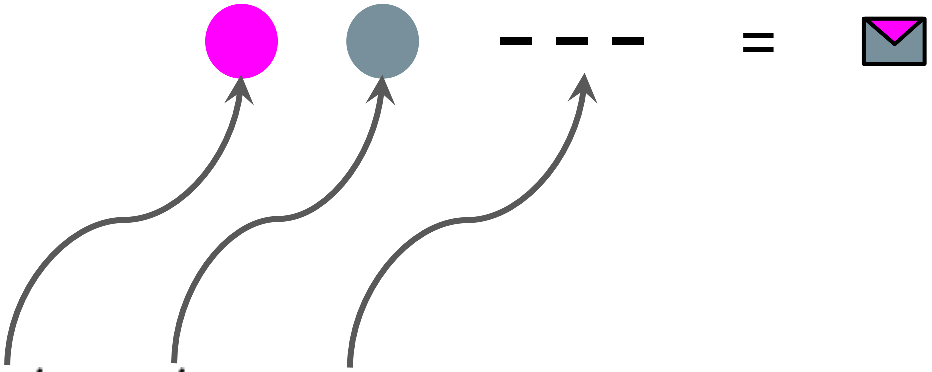


$$M(h_v^t, h_w^t, e_{vw})$$

Message passing

Message function:

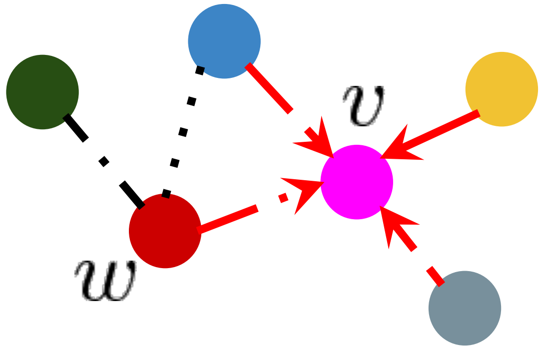


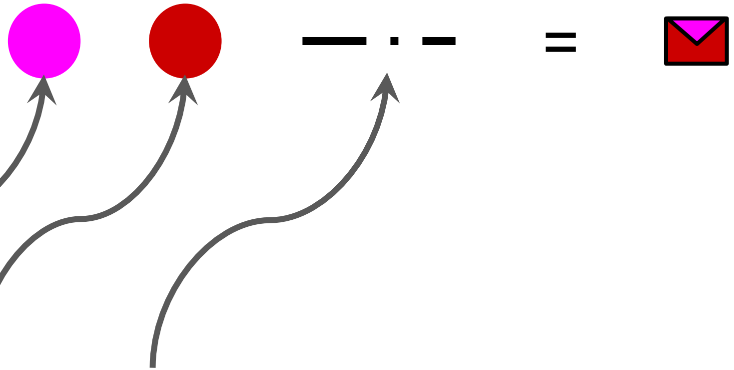


$$M(h_v^t, h_w^t, e_{vw})$$

Message passing

Message function:

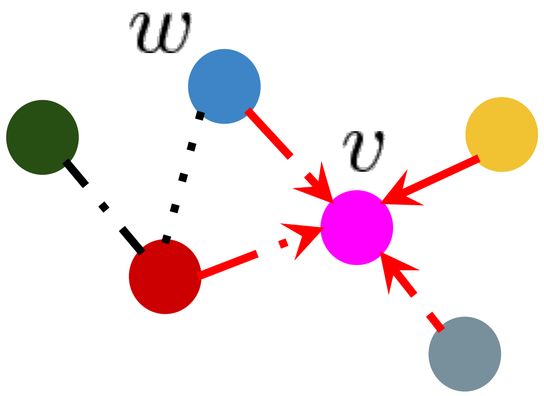


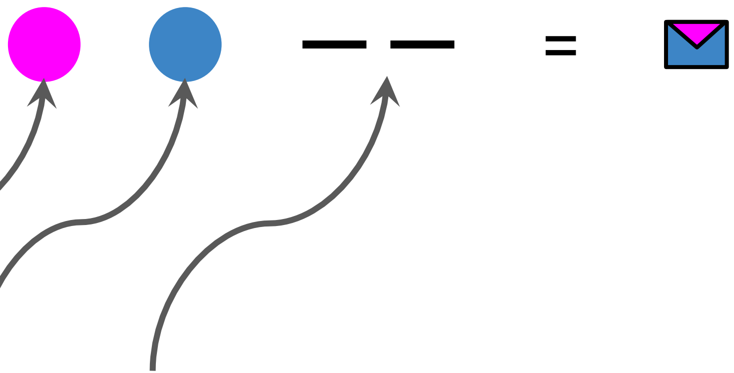


$$M(h_v^t, h_w^t, e_{vw})$$

Message passing

Message function:

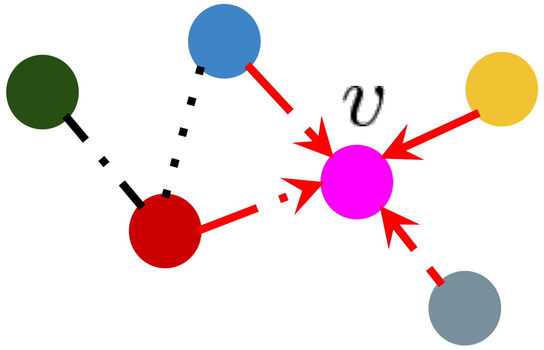


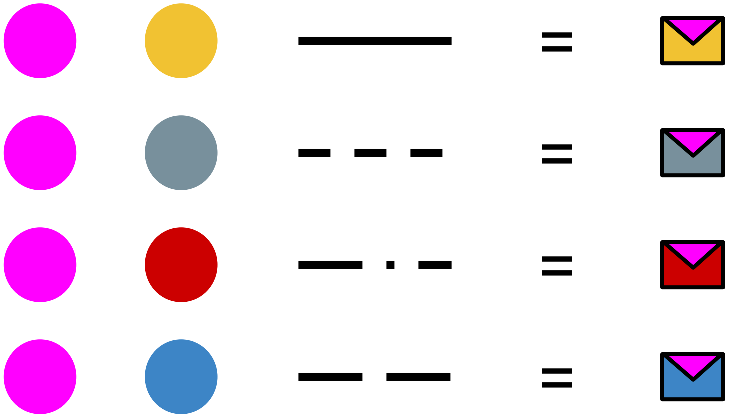


$$M(h_v^t, h_w^t, e_{vw})$$

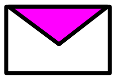
Message passing

Message function:



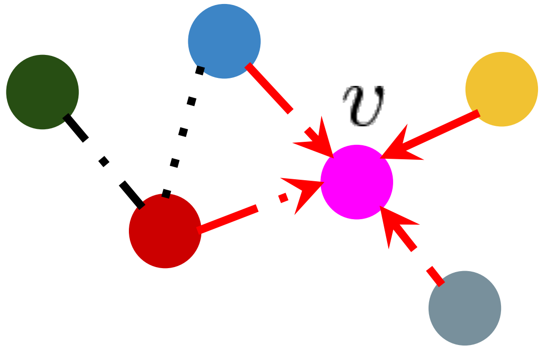


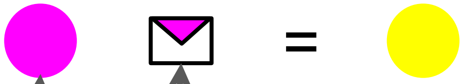
$$m_v^{t+1} = \sum_{w \in \mathcal{N}_v} M(h_v^t, h_w^t, e_{vw}) =$$



Message passing

Update function:

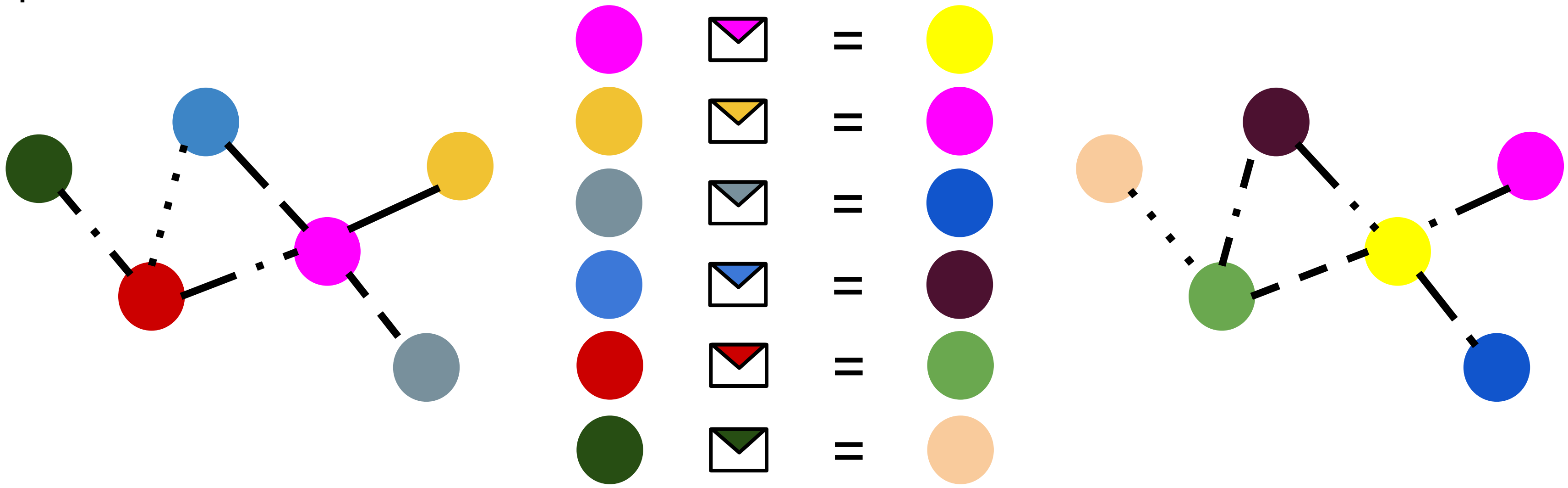




$$U(h_v^t, m_v^{t+1})$$

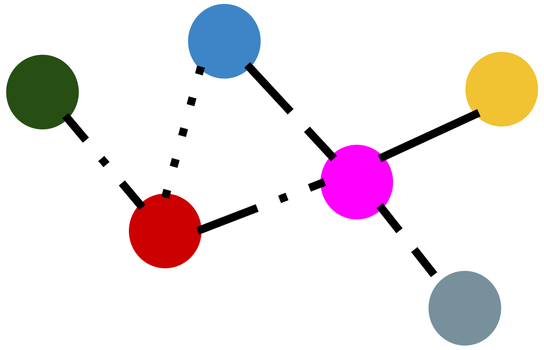
Message passing

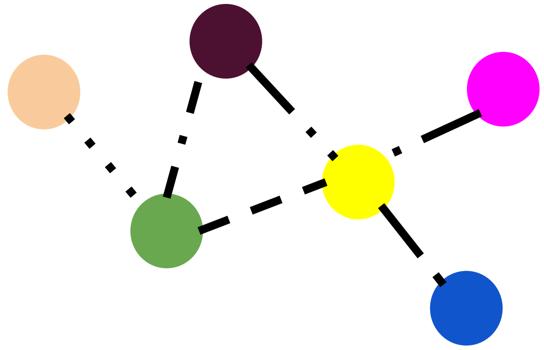
Update function:



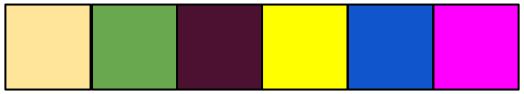
Readout

Readout function:





$$\hat{y} = R(\{h_v^T \mid v \in G\})$$ =



Convolutional Networks on Graphs

- Message Function

$$m_v^{t+1} = M(h_v^t, h_w^t, e_{vw}) = (h_w^t, e_{vw})$$

- Update Function

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) = \sigma(H_t^{deg(v)} m_v^{t+1})$$

- Readout Function

$$\hat{y} = R = f(\sum_{v,t} \mathrm{softmax}(W_t h_v^t))$$

where $(.,.)$ denotes concatenation, H_t^N are learned matrices one for each time step t and degree edge label, f is a neural network and σ is a non-linearity function such as ReLU

D. Duvenaud, D. Maclaurin, J. A. Iparraguirre, R. G. Bombarelli, T. Hirzel, A. Aspuru-Guzik, R. P. Adams. Convolutional Networks on Graphs for Learning Molecular Fingerprints, NIPS 2015

Gated Graph Sequence Neural Networks

- Message Function

$$m_v^{t+1} = M(h_v^t, h_w^t, e_{vw}) = A_{e_{vw}} h_w^t$$

- Update Function

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) = \mathrm{GRU}(h_v^t, m_v^{t+1})$$

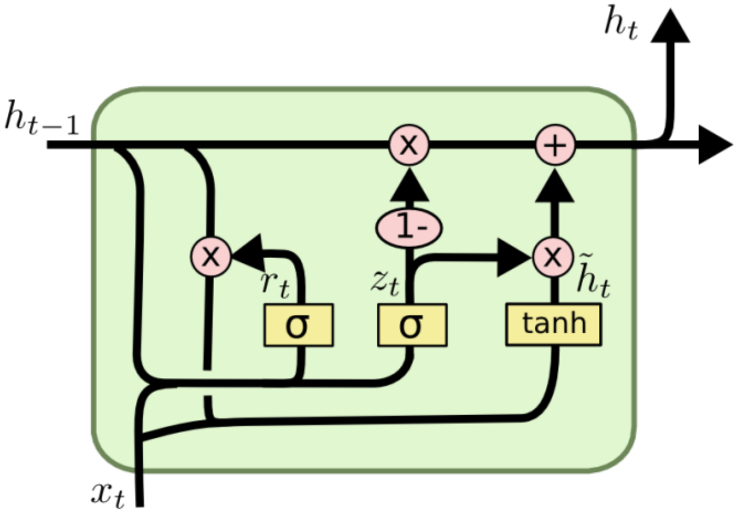
- Readout Function

$$\hat{y} = R = \sum_v \sigma(i(h_v^t, h_v^0) \odot (j(h_v^t))$$

where $A_{e_{vw}}$ is a learned matrix one for each discrete edge label, GRU is Gated Recurrent Unit, i, j are neural networks and $\odot$ is elementwise multiplication, σ is a non-linearity function such as ReLU

Y. Li, D. Tarlow, M. Brockschmidt and R. Zemel. Gated graph sequence neural networks. ICLR, 2016.

GRU



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Interaction Networks

- Message Function

$$m_v^{t+1} = M(h_v^t, h_w^t, e_{vw}) = f(h_v^t, h_w^t, e_{vw})$$

- Update Function

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) = g(h_v^t, x_v, m_v^{t+1})$$

- Readout Function

$$\hat{y} = R = f(\sum_v h_v^t)$$

where f, g represent neural networks, $(.,.)$ denotes concatenation, x_v is an external vector representing some outside influence to the node v.

P. W. Battaglia, R. Pascanu, M. Lai, D. Rezende, K. Kavukcuoglu. Interaction Networks for Learning about Objects, Relations and Physics, NIPS, 2016.

Molecular Graph Convolutions

- Message Function

$$m_v^{t+1} = M(h_v^t, h_w^t, e_{vw}) = e_{vw}^t$$

- Update Function

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) = \alpha(W_1(\alpha(W_0 h_v^t), m_v^{t+1}))$$

- Readout Function

$$\hat{y} = R = \alpha(W_4(\alpha(W_2, e_{vw}^t), \alpha(W_3(h_v^t, h_w^t))))$$

where $(.,.)$ denotes concatenation, W_i are learned weight matrices, α is the ReLU activation.

S. Kearnes, K. McCloskey, M. Berndl, V. Pande, P. Riley, Molecular Graph Convolutions: Moving Beyond Fingerprints, JCAMD, vol. 30, no. 8, 2016.

Convolutional and Locally Connected Neural Networks

- Message Function

$$m_v^{t+1} = M(h_v^t, h_w^t, e_{vw}) = C_{vw}^t h_w^t$$

- Update Function

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) = \sigma(m_v^{t+1})$$

where C_{vw} are parameterized by the eigenvectors of the graph Laplacian L and the other parameters of the model, σ is a non-linearity function such as ReLU

1. Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering, NIPS 2016.
2. Bruna et al., Spectral Networks and Locally Connected Networks on Graphs, ICLR 2014.

Graph Convolutional Networks

- Message Function

$$m_v^{t+1} = M(h_v^t, h_w^t, e_{vw}) = \frac{A_{vw}}{\sqrt{\deg(v)\deg(w)}} h_w^t$$

- Update Function

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) = \sigma(W^t m_v^{t+1})$$

where A_{vw} is a learnable parameter, W^t are learned matrices one for each time step, σ is a non-linearity function such as ReLU

T. Kipf and M. Welling. Semi-Supervised Classification with Graph Convolutional Networks, ICLR 2017.

Bibliography

Tutorials:

- Geometric Deep Learning, Tutorial, CVPR, 2017. http://geometricdeeplearning.com/
- Deep Learning on Graphs with Graph Convolutional Networks. http://deeploria.gforge.inria.fr/thomasTalk.pdf

List of papers:

- Gilmer et al., Neural Message Passing for Quantum Chemistry, 2017. https://arxiv.org/abs/1704.01212
- Kipf et al., Semi-Supervised Classification with Graph Convolutional Networks, ICLR 2017. https://arxiv.org/abs/1609.02907
- Defferrard et al., Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering, NIPS 2016. https://arxiv.org/abs/1606.09375
- Bruna et al., Spectral Networks and Locally Connected Networks on Graphs, ICLR 2014. https://arxiv.org/abs/1312.6203
- Duvenaud et al., Convolutional Networks on Graphs for Learning Molecular Fingerprints, NIPS 2015. https://arxiv.org/abs/1509.09292
- Li et al., Gated Graph Sequence Neural Networks, ICLR 2016. https://arxiv.org/abs/1511.05493
- Battaglia et al., Interaction Networks for Learning about Objects, Relations and Physics, NIPS 2016. https://arxiv.org/abs/1612.00222
- Kearnes et al., Molecular Graph Convolutions: Moving Beyond Fingerprints, 2016. https://arxiv.org/abs/1603.00856

Bibliography

Source Code / Repositories:

- Neural Message Passing for Computer Vision: https://github.com/priba/nmp_qc
- Graph Convolutional Networks in TensorFlow: https://github.com/tkipf/gcn
- Graph Convolutional Networks in PyTorch: https://github.com/tkipf/pygcn
- PyTorch implementation of graph ConvNets: https://github.com/xbresson/graph_convnets_pytorch
- Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering: https://github.com/mdeff/cnn_graph

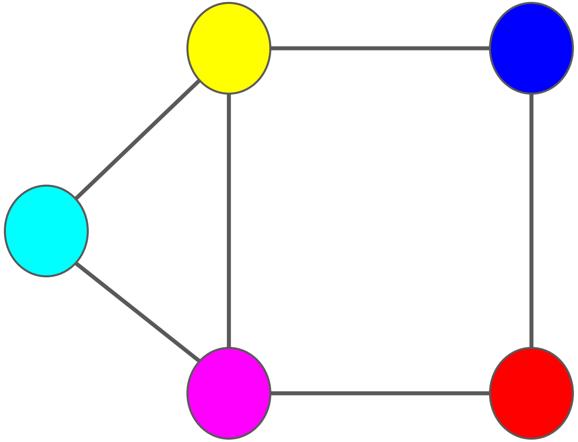
Summary: Neural network on graphs

- Success story of deep learning
- Convolutional Neural Network
    - Message passing
- Message passing in graphs
- Convolutional Neural network on graphs
- Graph convolutional networks

Programming languages, evaluation protocols, datasets and Programming Hands-on: Graph classification with RW kernel

Working with graphs

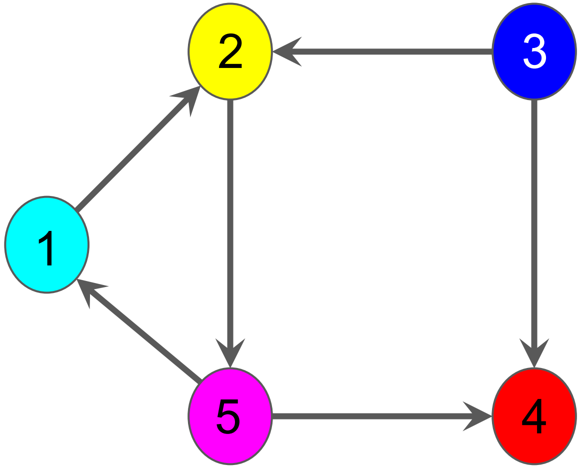
- Type of graph representation in computer memory
- There are two ways:
  - Sequential representation
  - Linked representation



Working with graphs

Sequential representation

- Adjacency matrix

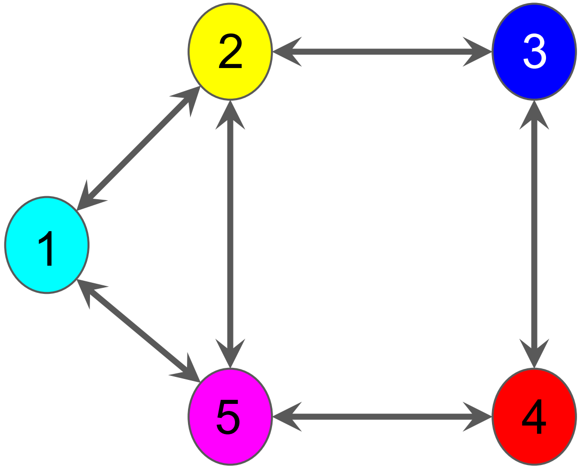


$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Working with graphs

Sequential representation

- Adjacency matrix

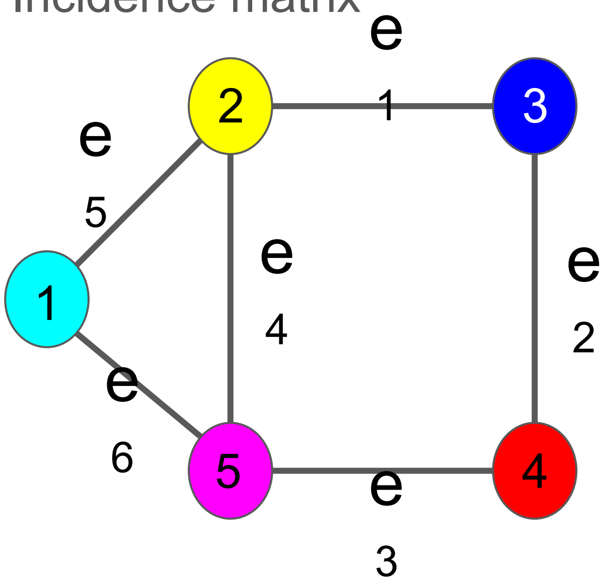


$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Working with graphs

Sequential representation

- Incidence matrix

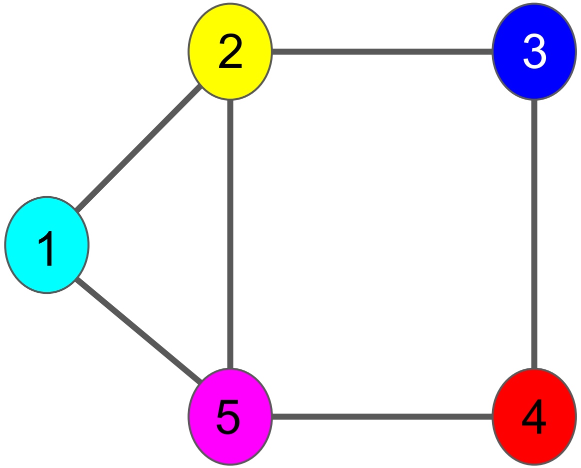


$$I = \begin{array}{c} \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{matrix} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

Working with graphs

Linked representation

- Adjacency list



$$A = \begin{array}{llll} \mathbf{1} & 2 & 5 & \\ \mathbf{2} & 1 & 3 & 5 \\ \mathbf{3} & 2 & 4 & \\ \mathbf{4} & 3 & 5 & \\ \mathbf{5} & 1 & 2 & 4 \end{array}$$

Programming languages

Matlab

MatlabBGL, Graph and Network algorithms, GAIMC

Python

Networkx, igraph

C/C++

Boost Graph Library

Evaluation protocols, tools and datasets

- MUTAG, PTC, ENZYMES, D&D, NCI1 and NCI109
- IAM graph database
- GREYC
- SESYD Graphs
- POLY-LINE
- ICPR 2016: SSGCI dataset
- ICPR 2016 : Graph Distance Contest

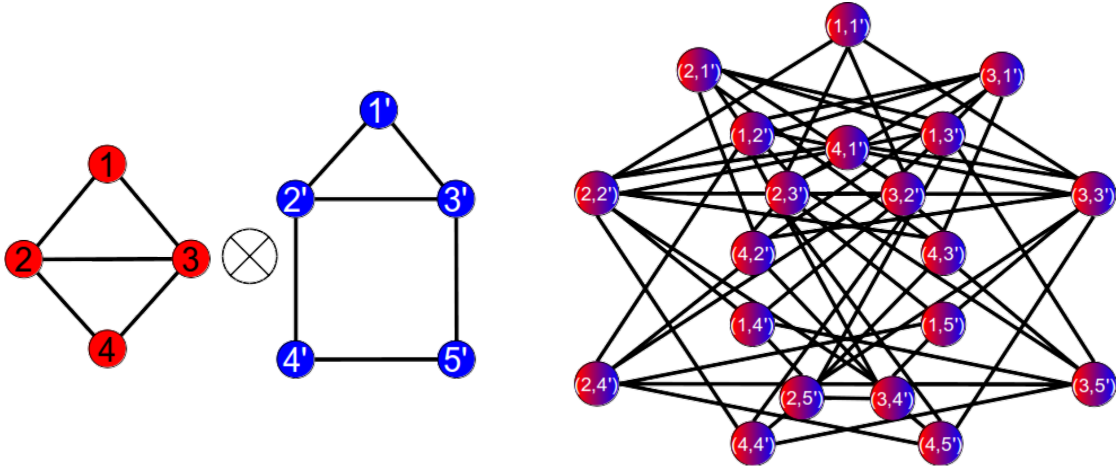
Programming Hands-on:
Graph classification with RW kernel

Random Walks on Graph

- Compare two graphs in terms of number of matching random walks.
- Discount contribution of larger walks because repeatability.
- Two graphs are similar if they have many matching walks, otherwise, dissimilar.
- Random walk on tensor product graph is equivalent to simultaneous random walk on input graphs.

1. H. Kashima, K. Tsuda and A. Inokuchi. “Marginalized kernels between labeled graphs”. ICML, 2003.
2. T. Gärtner, P. Flach, and S. Wrobel. “On graph kernels: Hardness results and efficient alternatives”. COLT, 2003.
3. S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor and K. M. Borgwardt. “Graph kernels”. JMLR, 2010.

Product Graph



$$V_X = \{(u_i, u_j) : u_i \in V_1, u_j \in V_2\}$$

$$E_X = \{((u_i, u_j), (v_i, v_j)) : (u_i, v_i) \in E_1 \wedge (u_j, v_j) \in E_2, u_i, v_i \in V_1, u_j, v_j \in V_2\}$$

Random Walks on Graph

- Walks of length *k* can be computed by looking at the *k*th power of the adjacency matrix.
- Summing the discounted exponential of the product graph results in the kernel.
- Normalization of the adjacency matrices of the input graphs is crucial.

1. H. Kashima, K. Tsuda and A. Inokuchi. “Marginalized kernels between labeled graphs”. ICML, 2003.
2. T. Gärtner, P. Flach, and S. Wrobel. “On graph kernels: Hardness results and efficient alternatives”. COLT, 2003.
3. S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor and K. M. Borgwardt. “Graph kernels”. JMLR, 2010.

Random Walks on Graph

- Kernel definition:

$$K(\mathcal{G},\mathcal{G}') = \frac{1}{|\mathcal{G}||\mathcal{G}'|}\sum_k \frac{\lambda^k}{k!}\mathbf{e}^T A_\times^k \mathbf{e}$$
$$= \frac{1}{|\mathcal{G}||\mathcal{G}'|}\mathbf{e}^T \exp(\lambda A_\times)\mathbf{e}$$
$$= \frac{1}{|\mathcal{G}||\mathcal{G}'|}\mathbf{e}^T (\mathbf{I} - \lambda A_\times)^{-1}\mathbf{e}$$

- $(\mathbf{I} - \lambda A_\times)^{-1}\mathbf{e}$ can be obtained by solving the linear systems $(\mathbf{I} - \lambda A_\times)x = \mathbf{e}$

MUTAG dataset

- Graphs representing chemical compounds which are mutagenic or non-mutagenic.
- Total 188 graphs of 2 classes (125 vs 63).
- Small but unbalanced dataset.

GMPRDIA.ipynb

System configuration

- Ubuntu 16.04
- Python 3.5

Prerequisite packages

- networkx
- nxpd
- os
- numpy
- scipy
- glob
- sklearn

Discussion

12h15 – 12h30

Discussion

- Are graphs still relevant?
- Are graph-based methods still useful for Pattern Recognition and Document Image Analysis?
- Modern trends in CNN and traditional structural methods?
- Do/have you use(d) graphs?
- Are you motivated to use them in future?