東京工業大学
Tokyo Institute of Technology

MURATA LAB

Mid-term Presentation

Motif-aware method for graph analysis

Hoang Nguyen (M2), Supervisor: Assoc. Prof. Tsuyoshi Murata

2017/02/03

Roadmap

Result of motif methods | Proposal of future models

NETWORK MOTIF
Definition
Applications
Motif Conductance

Deep architecture
Skip-gram methods
Spectral methods
EMBEDDINGS

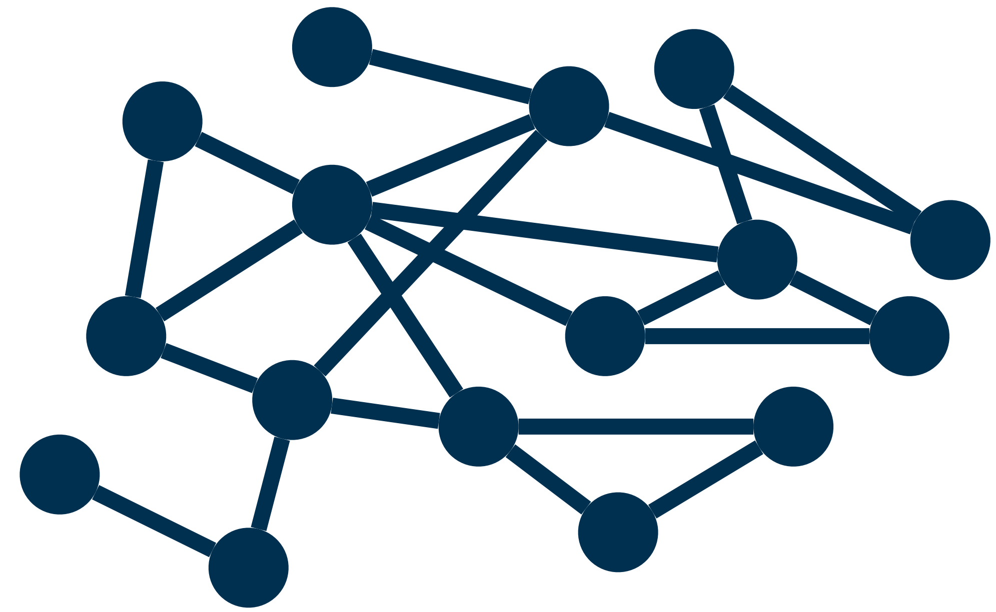
OBSERVATION
Experiments
Proposed models
Results

Diversity models
Generative models
Motif Spectral
IN PROGRESS

Definition and applications
Complex networks and significant subgraphs

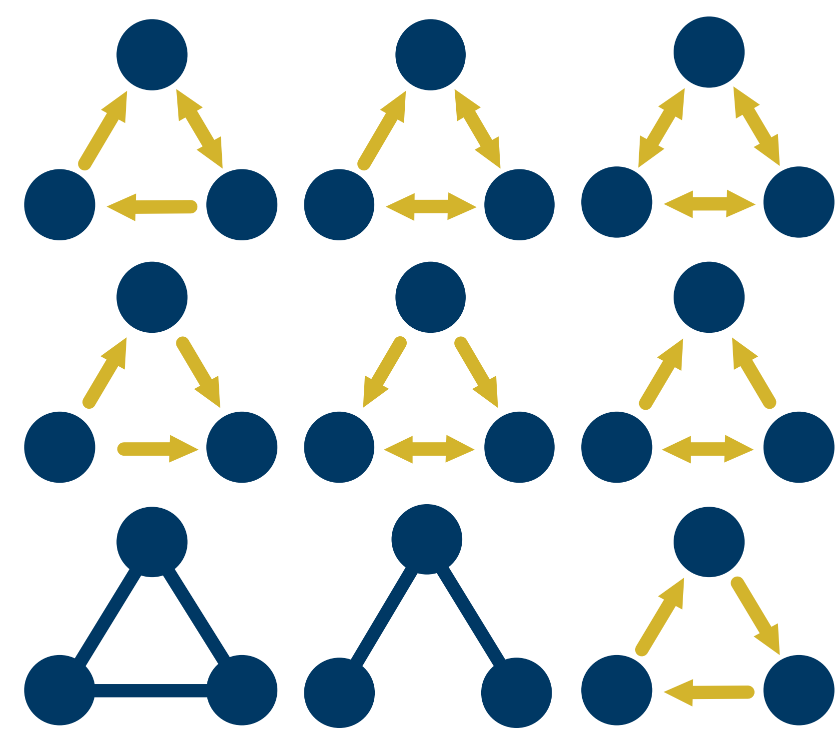
Scales of network analysis:

- Macroscopic (global view)
- Microscopic (interactions)
- Mesoscopic (sub-structures)



Network motifs are [1]

> patterns of statistically significant interconnections occurring in complex networks.



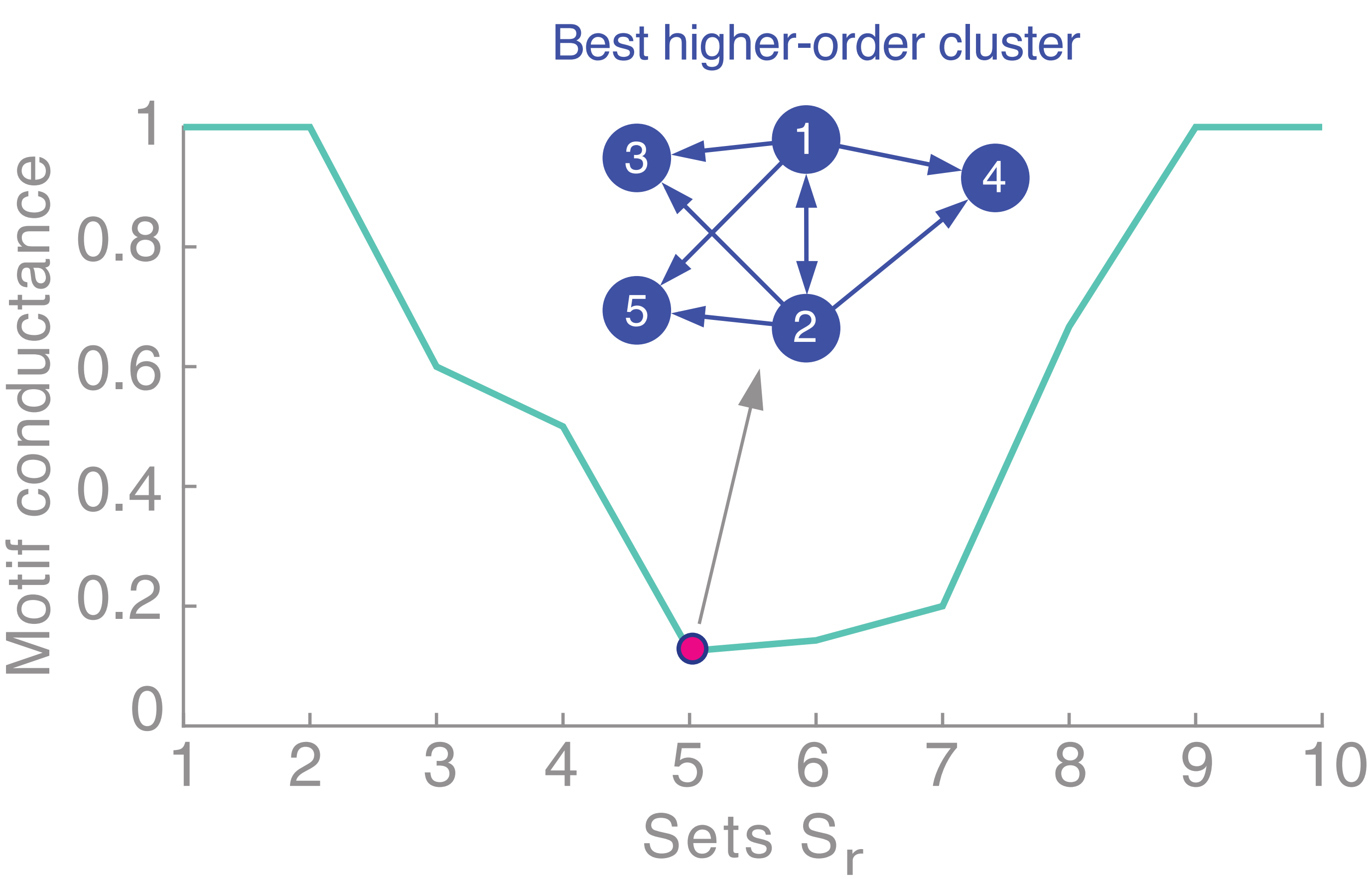
Applications

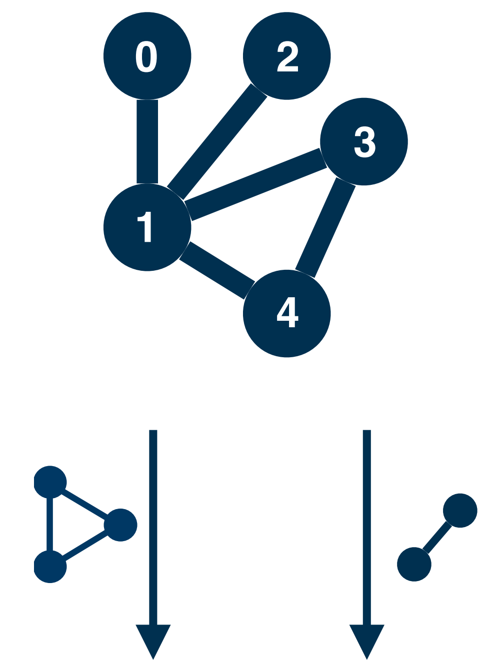
- Social networks [2]
- Biological systems [1,3,4]

[4](Franceschet, 2012) [3](Honey, 2007) [2](Holland, 1970) [1](Milo, 2002)

Motif conductance *(Benson, 2016)*

- Generalization of graph conductance.
- Motif-cooccurrence matrix.
- Good results for graph partitioning.



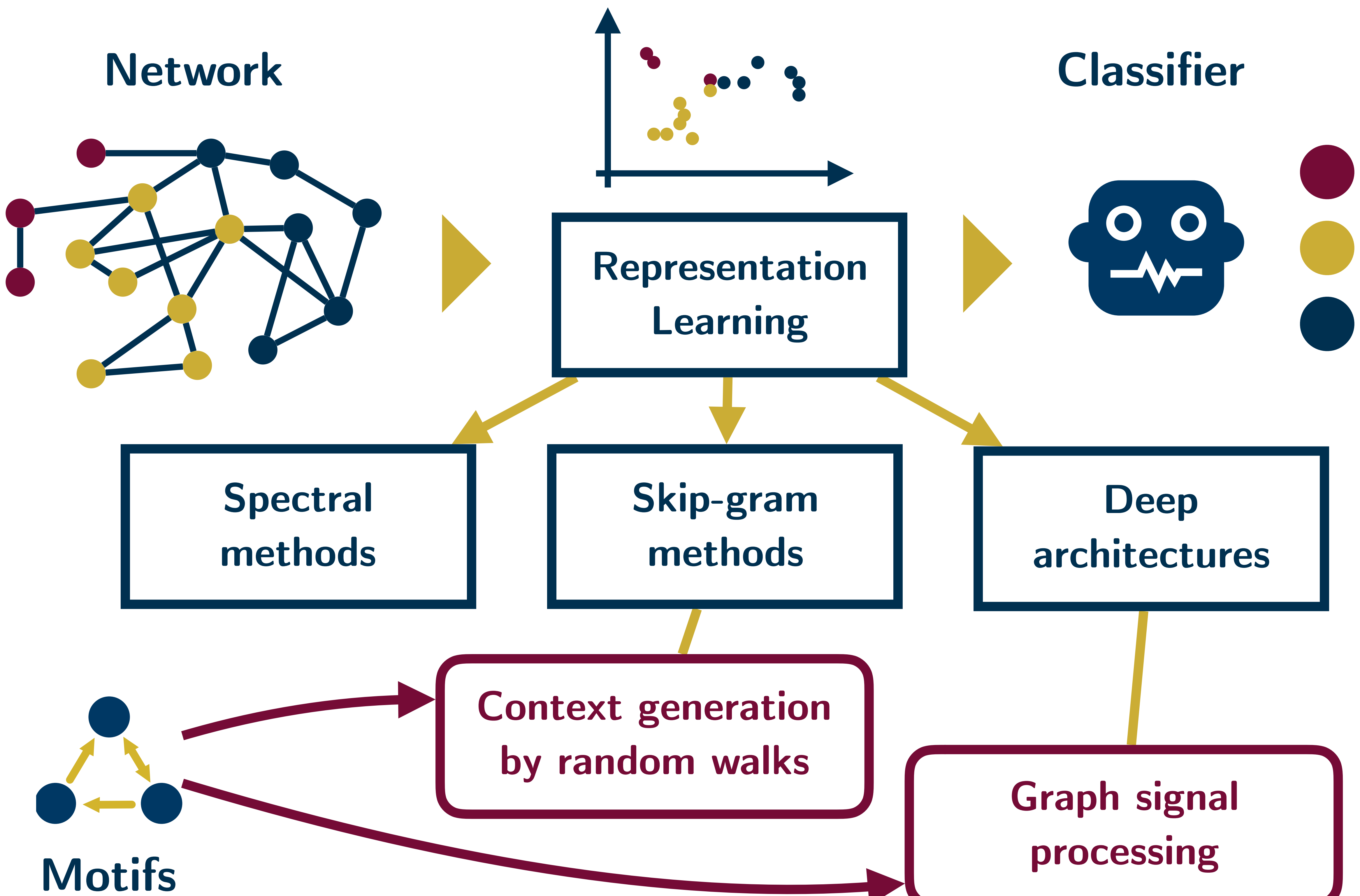


$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 2 & 2 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 2 \\ 0 & 2 & 0 & 2 & 0 \end{bmatrix}$$

$$\phi_m(S) = \frac{\mathrm{cut}_M(S, \bar{S})}{\min[\mathrm{vol}_M(S), \mathrm{vol}_M(\bar{S})]}$$

Experiments

Graph embedding for classifying network node



Datasets:

- Blogcatalog3
- Cora, Citeseer, Pubmed
- Facebook (EgoNet)
- Transcription networks

Networks	***#nodes***	***#edge***
Blogcatalog3	10312	333983
Cora*	2708	5429
Citeseer*	3327	4732

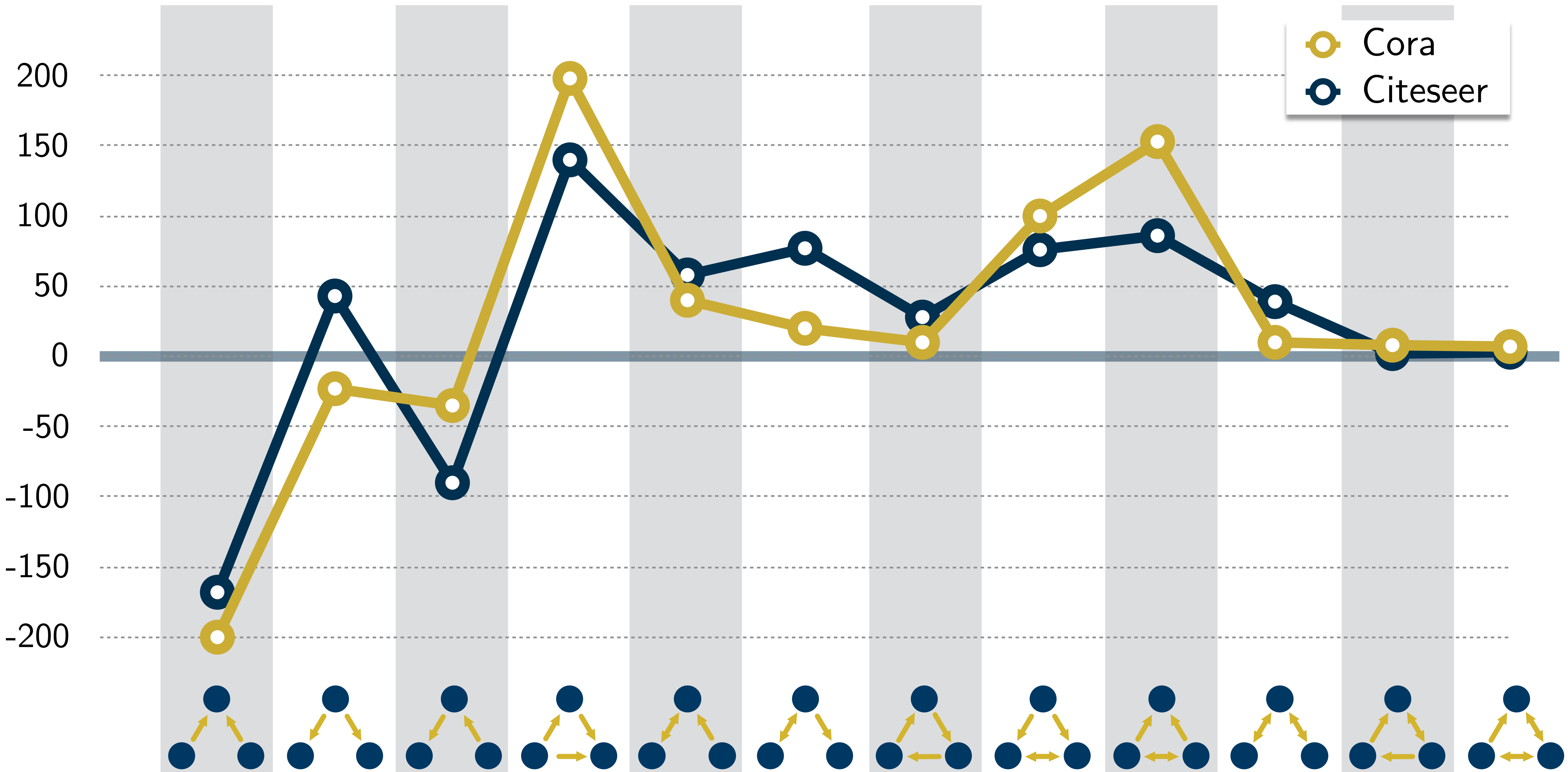
* Each node has a feature vector (tf-idf)

Deciding motif:

- Motif frequency
- Z-score
- Motif conductance

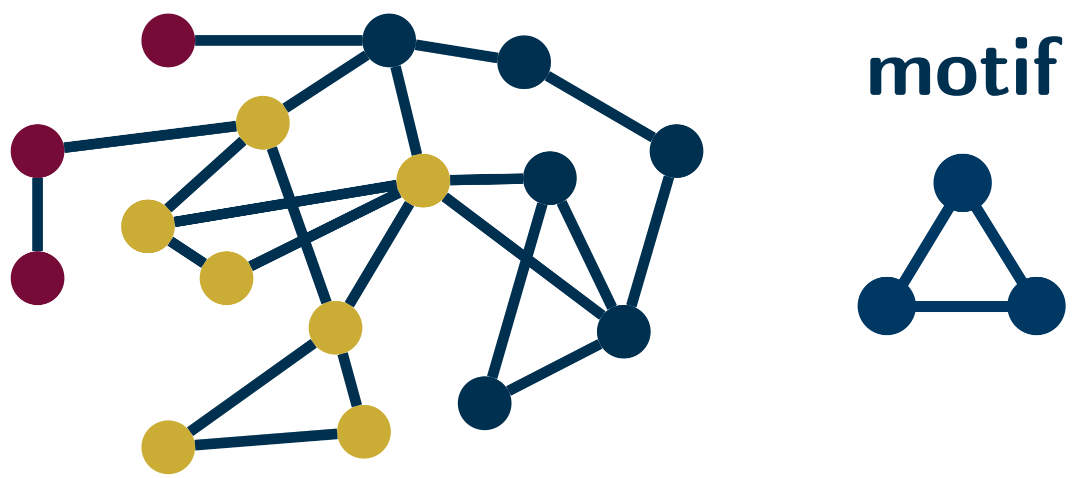
$$z_m = \frac{N_m(emp) - N_m(rnd)}{\sigma_m}$$

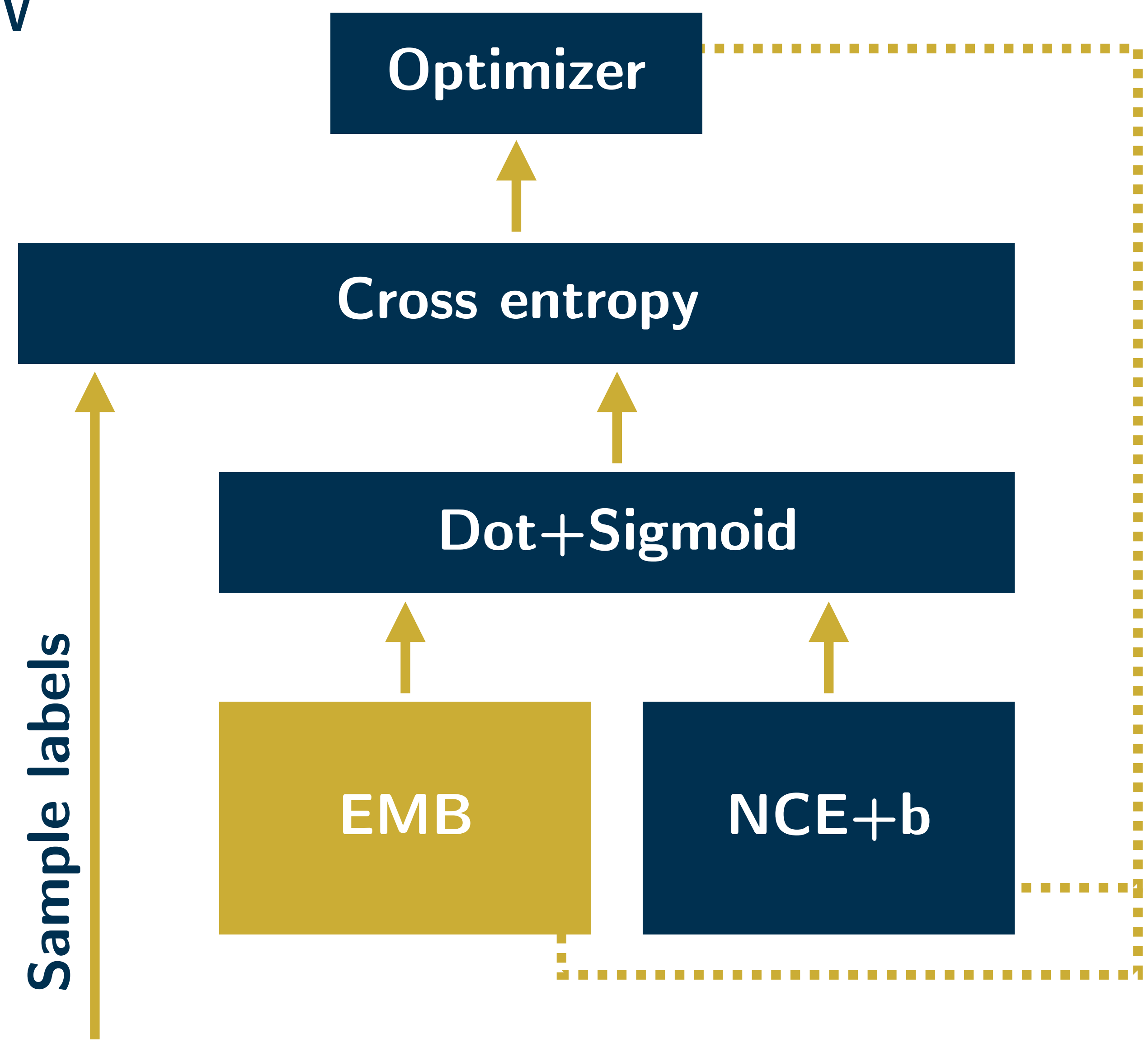
Significant graph - Directed size 3 motifs



Using motif as a guiding pattern

- DeepWalk := Skip-gram+RW
- LINE := DeepWalk+bi-fan
- node2vec := Skip-gram +bias-ed walk
- motifwalk := Skip-gram +motif walk

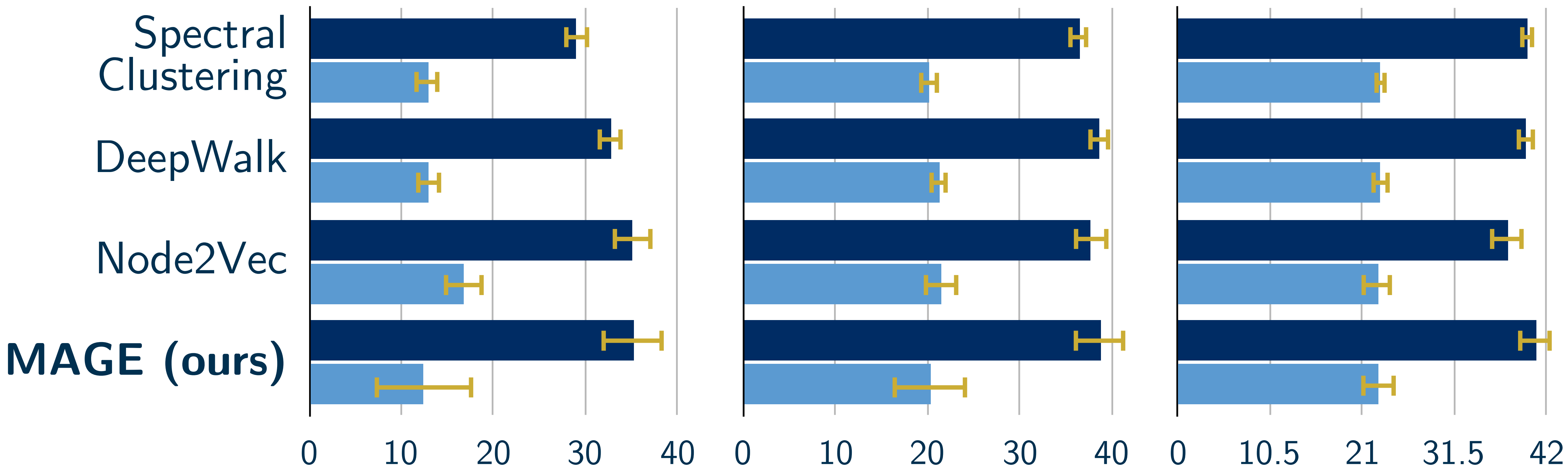




Generated context:

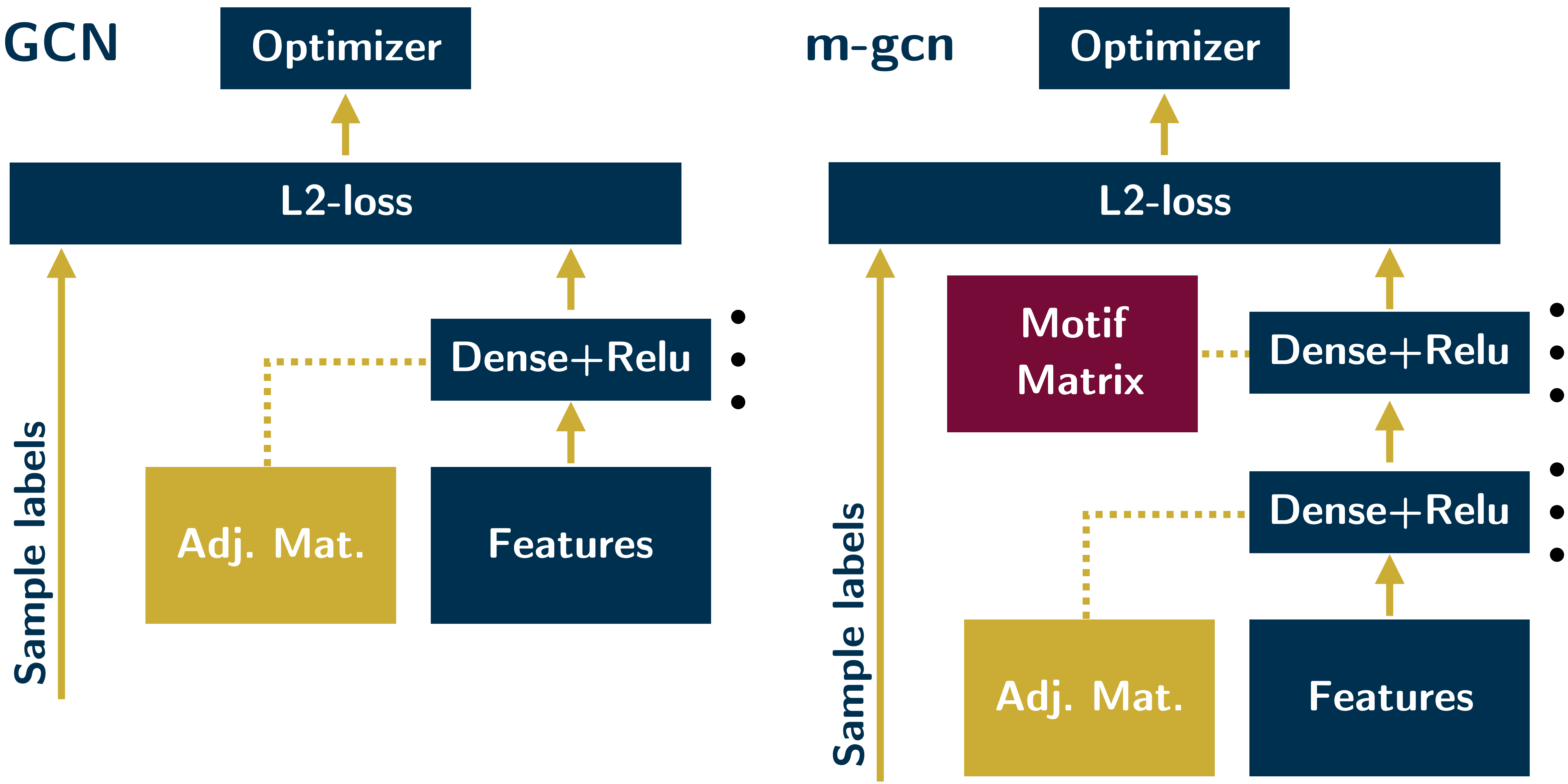
Node multi-label classification

- Blogcatalog3 - f1 score
- 39 classes (imbalanced), nodes can have multiple labels



- Bottleneck at context generation.
- The optimization process is inefficient.

Motif laplacian for graph convolutional networks



Node label classification

- Citation networks: Cora, citeseer, pubmed. - f1 and acc.
- 3 to 6 classes, each node has exactly one label.

Methods	Cora		Citeseer		Pubmed	
	accuracy	macro	accuracy	macro	accuracy	macro
Spectral Clus.	59.5	-	42.7	-	65.3	-
DeepWalk	67.2	-	43.2	-	65.3	-
GCN	81.5	-	70.3	-	79.0	-
m-gcn (ours)	**83.3**	-	**71.4**	-	**80.4**	-

- Explaining why does it work.
- Minor contribution.

My interests:

- Network sampling (diversity oriented).
- Turn m-gcn into a generative model.

Agenda:

- Investigate graph spectral theory.
- Determinantal processes on graphs.
- Graph motif sampling model definition.
- (opt.) Generative Adversarial Networks and Auto-Encoders.

References

Network Science | Machine Learning

(Diestel, 2000) Graph Theory.

(Mikolov, 2013) Distributed Representations of Words and Phrases and their Compositionality.

(Bengio, 2013) Deep Learning of Representations: Looking Forward.

(Ng, 2001) On Spectral Clustering: Analysis and an algorithm.

(Perozzi, 2014) DeepWalk: Online Learning of Social Representations.

(Tang, 2015) LINE: Large-scale Information Network Embedding.

(Cao, 2015) GraRep: Learning Graph Representations with Global Structural Information.

(Grover, 2016) node2vec: Scalable Feature Learning for Networks

(Yang, 2016) Semi-Supervised Classification with Graph Convolutional Networks

(Alon, 2007) Network motifs: theory and experimental approaches.

(Benson, 2016) Higher-order organization of complex networks.

Motif generalizes link

Motif here!

Skip-gram model

Applying NLP techniques to network science

Algorithm 1: Motif context sampling $P_m(\omega)$.

Data: Undirected Graph $G = (V, E)$;
Input: start vertex, walk length, motif $M = (V', E')$, bias β;
Result: Vertices in the local motif structure;

```
vertexList ⟵ [start vertex]
current ⟵ start vertex
initialize dequeue buffer Q of size |V'|
while vertexList.length < walk length do
    candidate ⟵ random(neighboor(current))
    Q.append(candidate)
    if Isomorphic(Q, M) then
        if random < β then
            vertexList.append(candidate)
        else
            continue
    else
        if random > β then
            vertexList.append(candidate)
        else
            continue
return vertexList
```

NPL Skipgram

Laplacian, normalized, convolution operations