subgraph2vec

Article

Narayanan A, Chandramohan M, Chen L, et al. subgraph2vec: Learning Distributed Representations of Rooted Sub-graphs from Large Graphs[J]. 2016.

Data

Aim:

Learning latent representations of rooted subgraphs from large graphs.

Contributions:

- Propose subgraph2vec.
- Develop a modified version of the skipgram language model.
- Discuss how subgraph2vec's representation learning technique would help to build the deep learning variant of WL kernel.

Applications:

- graph classification and clustering tasks
- code clone detection
- malware detection

Data

Key results:

- Benchmark Datasets

Table 3: Average Accuracy (± std dev.) for subgraph2vec and state-of-the-art graph kernels on benchmark graph classification datasets

Dataset	MUTAG	PTC	PROTEINS	NCI1	NCI109
WL [6]	80.63 ± 3.07	56.91 ± 2.79	72.92 ± 0.56	80.01 ± 0.50	80.12 ± 0.34
Deep WL_{YV} [7]	82.95 ± 1.96	59.04 ± 1.09	73.30 ± 0.82	**80.31** ± 0.46	**80.32** ± 0.33
subgraph2vec	**87.17** ± 1.72	**60.11** ± 1.21	**73.38** ± 1.09	78.05 ± 1.15	78.39 ± 1.89

- Clone Detection

Table 5: Clone Detection - Results

Kernel	WL [6]	Deep WL_{YV} [7]	subgraph2vec
Pre-training duration	-	421.7 s	409.28 s
ARI	0.67	0.71	**0.88**

- Malware Detection

Table 7: Malware Detection - Results

Classifier	WL [6]	Deep WL_{YV} [7]	subgraph2vec
Pre-training duration	-	2631.17 s	2219.28 s
Accuracy	66.15	71.03	**74.48**

Background:

- language model

Methods:

- Generate rooted subgraphs: WL Relabeling Process
- Embeddings of those subgraphs: Radial Skip Gram Model
- Approximate the probability distribution: Negative Sampling

➢ **for** *each* $sg_{cont} \in context_v^{(d)}$ **do**

$$J(\Phi) = -\log \Pr\left(sg_{cont} | \Phi(sg_v^{(d)})\right)$$

Comments or Questions

- When subgraph of degree 0 are considered, subgraph2vec provides node embeddings.
- How can we get the initial labels for each node? ☑
- What is the Deep WL kernel? ☑

<table>
<tr><th colspan="4">graph matching</th></tr>
<tr><th>Article</th><th>Data</th><th>Data</th><th>Comments or Questions</th></tr>
<tr>
<td>Sussman Daniel, Park Youngser, Priebe Carey E, et al. Matched Filters for Noisy Induced Subgraph Detection.. 2019,</td>
<td>Aim:
Finding the vertex correspondence between two noisy graphs with different number of vertices where the smaller graph is still large.
Contributions:
● Proposed a number of padding methods to transform the noisy subgraph detection problem into a graph matching problem.

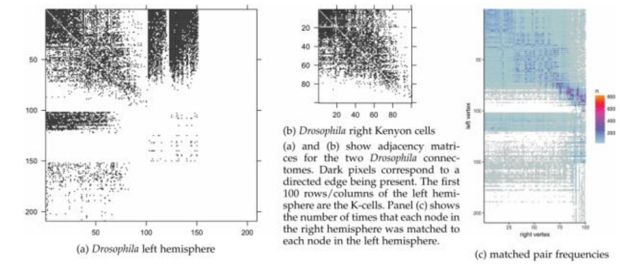
Applications:
● determining whether a particular activity pattern is present in a social network
● detecting certain shapes in an image
● discovering motifs in brain networks</td>
<td>Key results:
● Finding Kenyon Cells in a Drosophila Connectome


Fig. 5. Connectomes and matching analysis of the Drosophila connectomes.

Background:
● Subgraph Isomorphism
● Graph Matching Algorithms
● Statistical Models

Methods:
● Padding Approaches: Naive Padding, Centered Padding, Oracle Padding
● Graph Matching Approaches: FAQ algorithm</td>
<td>● 三种padding方法没看明白。。</td>
</tr>
</table>

Ontology matching

Article	Data	Data	Comments or Questions

Article

Mohammadi M，Hofman W，Tan Y H．A Comparative Study of Ontology Matching Systems via Inferential Statistics[J]. IEEE Transactions on Knowledge and Data Engineering, 2019, 31(4):615-628.

Data

Aim:

This paper examines the alignment systems using statistical inference since averaging is statistically unsafe and inappropriate.

Contributions:

- It is the first paper considering the statistical inference for comparison of two or more ontology matching systems.

Applications:

- The statistical methods are then applied to benchmark and multifarm tracks from the ontology matching evaluation initiative (OAEI) 2015 and their results are reported and visualized by critical difference diagrams.

Data

Key results:

- Comparison of two systems

TABLE 1
The Tests for Comparison of Two Systems over N Datasets

Test	Presumptions	Applicability
Paired t	Normality of differences	N > 30
Signed-rank	symmetry of differences w.r.t median	N > 10
McNemar	-	N < 10

Applicability is roughly the situation that test can be used and its results are valid and differences *refer to the differences in performance scores.*

- Comparison of multiple systems

TABLE 4
The Tests for Comparison of Multiple Systems over N Datasets

Test	Presumptions	Applicability
ANOVA t	Sphericity	N > 30
Friedman	-	N > 10
Quade	-	N < 10

Applicability is roughly the situation that test results are valid.

Background:

- Ontology matching
- Hypothesis testing

Methods:

- Comparison of two systems: Paired t-Test, Wilcoxon Signed-Rank Test, McNemar’s Test
- Comparison of multiple systems: Omnibus Tests, Post-Hoc Analysis

Comments or Questions

- 需要补一补概率论的假设检验部分