7th Joint International Conference on Data Science & Management of Data (CODS-COMAD 2024)

VeNoM: Approximate Subgraph Matching with Enhanced Neighbourhood Structural Information

Shubhangi Agarwal† Sourav Dutta†† Arnab Bhattacharya†

sagarwal@cse.iitk.ac.in sourav.dutta2@huawei.com arnabb@cse.iitk.ac.in

†Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, **India**

††Huawei Research Centre, Dublin, **Ireland**

Objectives

1. Approximate Subgraph Matching (ASM)
2. Study effects of neighborhood size on subgraph similarity computation

Motivation

Factors affecting ASM performance:

- Number of nodes
- Density of graph
- Label distribution
- Query size

.
.
.

- **Size of neighborhood matched** → **Depth of neighborhood (h)**, **Breadth of neighborhood (k)**
- Degree distribution

Parametrization - VeNoM-(k, h)

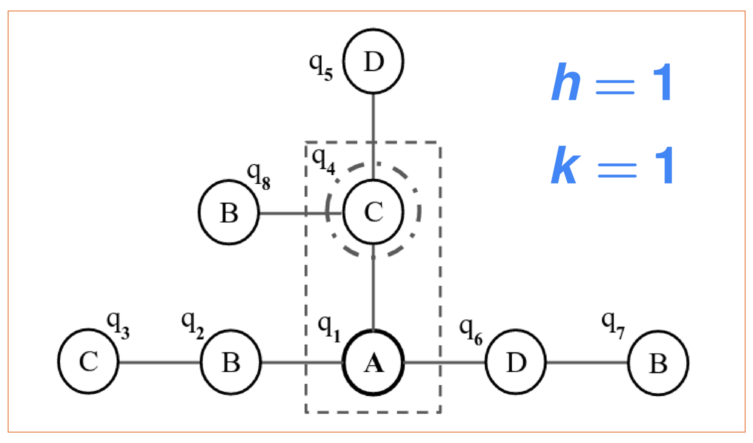
Unit

- Ordered collection of neighbor labels of a vertex
- Forms a path of length h

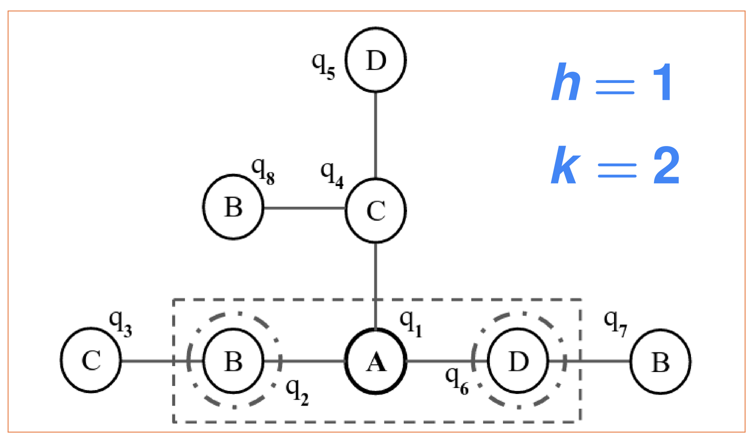
Group:

- Set of k units of a vertex along with its label
- Units correspond to different neighbors

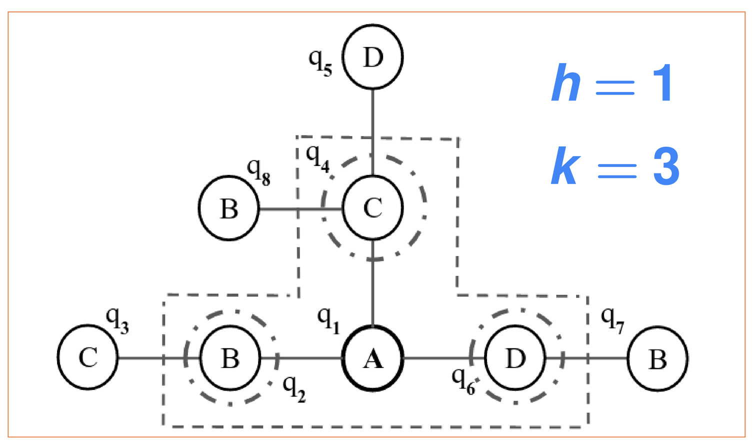
VeNoM-(1,1)



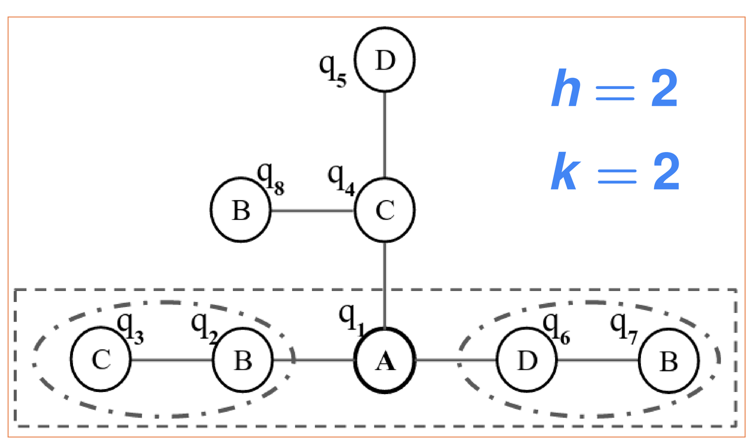
VeNoM-(2,1)



VeNoM-(3,1)



VeNoM-(2,2)



Stages of VeNoM

Step 1	Step 2	Step 3	Step 4
Neighbor information structuring	***Candidate set construction***	***Match Similarity Computation***	***Expand Partial Match***
Enumerate *units* and *groups*, labels, degree etc.	Prune search space (e.g., label similarity)	Pearson's χ^2 for statistical significance of match	Expand greedily with statistically significant matches

Statistical Significance

- Capture deviation of observed behavior from expected
- Pearson χ^2 statistic: $\chi^2 = \sum_i \frac{[O(s_i) - E(s_i)]^2}{E(s_i)}$
- Random match ⟶ Low similarity; Higher deviation $\implies$ Exceptional similarity

Similarity Computation - VeNoM-(2,2)

Candidate vertex groups: $\langle A, \langle B,C\rangle, \langle D,B\rangle \rangle$, $\langle A, \langle B,C\rangle, \langle C,B\rangle \rangle$, $\langle A, \langle B,C\rangle, \langle C,D\rangle \rangle$, $\langle A, \langle D,B\rangle, \langle C,B\rangle \rangle$, $\langle A, \langle D,B\rangle, \langle C,D\rangle \rangle$

Query group: $\langle A, \langle B,C\rangle, \langle B,B\rangle \rangle$

Match Categories:

s_0: $\langle A, \langle \times,\times\rangle, \langle \times,\times\rangle \rangle$ - $(u_0 \wedge u_0)$

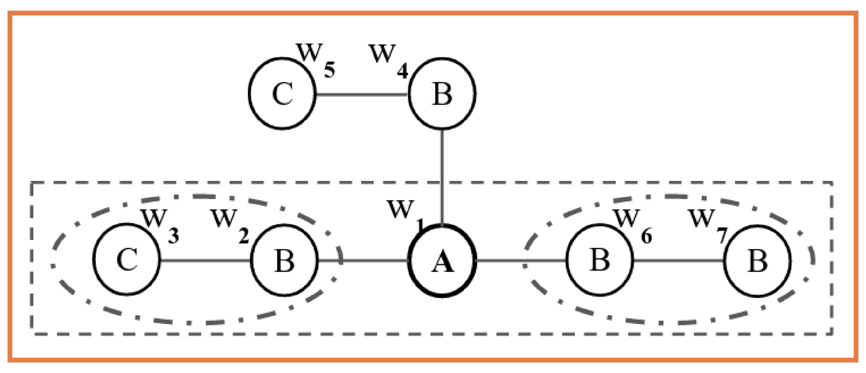
s_1: $\langle A, \langle B,\times\rangle, \langle \times,\times\rangle \rangle$ - $(u_0 \wedge u_1)$

s_2: $\langle A, \langle B,\times\rangle, \langle \times,B\rangle \rangle$ - $(u_0 \wedge u_2) \vee (u_1 \wedge u_1)$

s_3: $\langle A, \langle B,C\rangle, \langle \times,B\rangle \rangle$ - $(u_1 \wedge u_2)$

s_4: $\langle A, \langle B,C\rangle, \langle B,B\rangle \rangle$ - $(u_2 \wedge u_2)$

Example Query Graph



Query groups = Length of match symbol vector = 3

$O = \{s_3, s_2, s_2\} \implies \{s_0 : 0, s_1 : 0, s_2 : 2, s_3 : 1, s_4 : 0\}$

Expected behavior:

Dependent on degrees of query node and its 1-hop neighbors of and number of labels.

p_q = Probability of mismatch of 1-hop label

β_q = Probability of mismatch of 2-hop label

δ_q = Probability of mismatch of 2-hop label given 1-hop label mismatched

$P_q(u_0) = p_q \cdot \beta_q$; $P_q(u_1) = (\bar{p}_q \cdot \delta_q) + (p_q \cdot \bar{\beta}_q)$; $P_q(u_2) = \bar{p}_q \cdot \bar{\delta}_q$

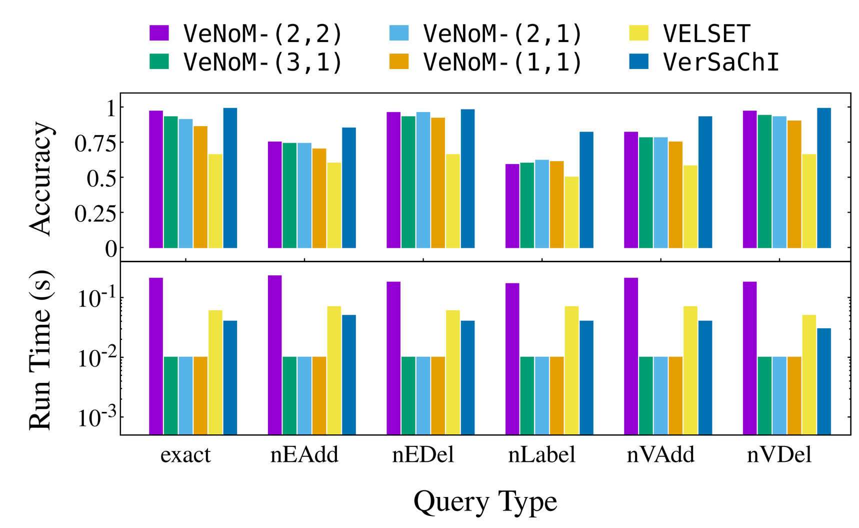
$P_q(s_3) = 2 \cdot \left(P_q(u_1) \cdot P_q(u_2)\right) = 2 \cdot \left((\bar{p}_q \cdot \delta_q) + (p_q \cdot \bar{\beta}_q)\right) \cdot (\bar{p}_q \cdot \bar{\delta}_q)$

Results - Real-world Graphs

Characteristics of Graphs

Dataset	#Vertices	#Edges	#Labels
Human	4.6K	86.2K	44
HPRD	9.4K	37K	307
Flickr	80.5K	5.9M	195
PPI	12K	10.74M	2.4K

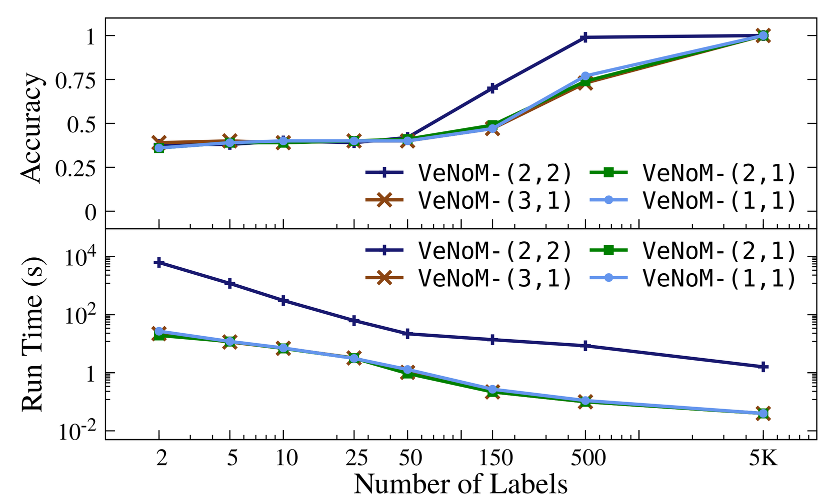
Performance on HPRD



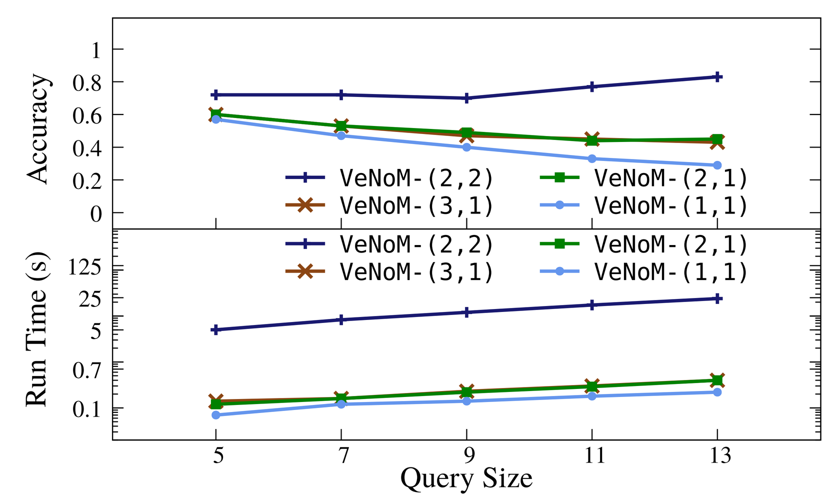
Results - Synthetic Barabási-Albert Graphs

Default Barabási-Albert graph parameters: $|V| = 100K$, $m = 50$, $l = 150$

Label Scaling



Query Size



- VeNoM-(2,2) more robust than counterparts due to structural look-ahead ability
- Ratio of label set size and node degree crucial

Key Observations

- Increase in depth of neighborhood $\implies$ trade-off between time and accuracy
- Increase in breadth with depth may increase runtime efficiency (larger group size may lead to lower number of groups)

References

[1] VerSaChI: Agarwal *et al*, CIKM 2021;

[2] VELSET: Dutta *et al*, WWW 2017

For more details

Code

Paper