Sampling Subgraphs with Guaranteed Treewidth for Accurate and Efficient Graphical Inference

Jaemin Yoo[1] jaeminyoo@snu.ac.kr

U Kang[1] ukang@snu.ac.kr

Mauro Scanagatta[2] mscanagatta@fbk.eu

Giorgio Corani[3] giorgio@idsia.ch

Marco Zaffalon[3] zaffalon@idsia.ch

[1]Seoul National University [2]Fondazione Bruno Kessler [3]IDSIA

Data Mining Laboratory · Seoul National University · USI/SUPSI IDSIA

OVERVIEW

- **Given** a large real-world graph $G = (V, E)$
- **Problem:** to sample a subgraph $U = (V, E')$ such that $E' \subseteq E$, which preserves the properties of G
- **Main idea:** to bound the treewidth of U to k
- **Source codes:** https://datalab.snu.ac.kr/btw

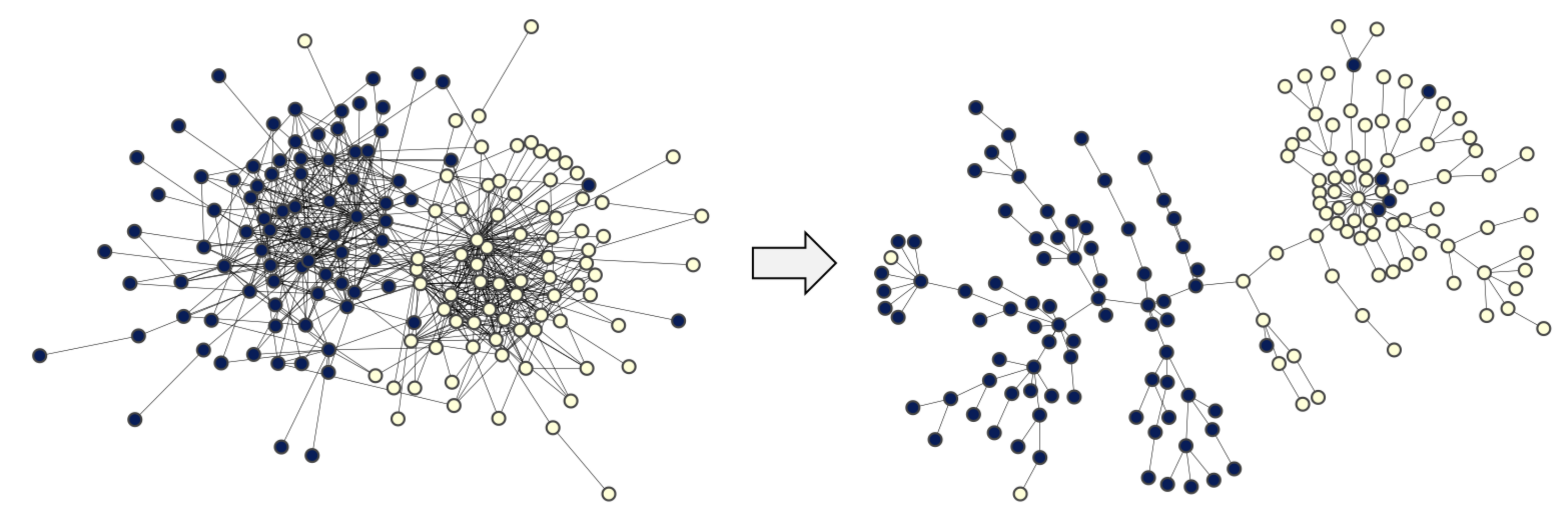
MOTIVATION

Graphical Inference

- Essential task for solving node classification
- **Loopy belief propagation (LBP)**
  - Fast but runs approximate inference
- **The junction tree algorithm (JT)**
  - Slow but runs exact inference
  - Exponential complexity with treewidth(G)
- **Treewidth:** how much G resembles a tree

Research Motivation

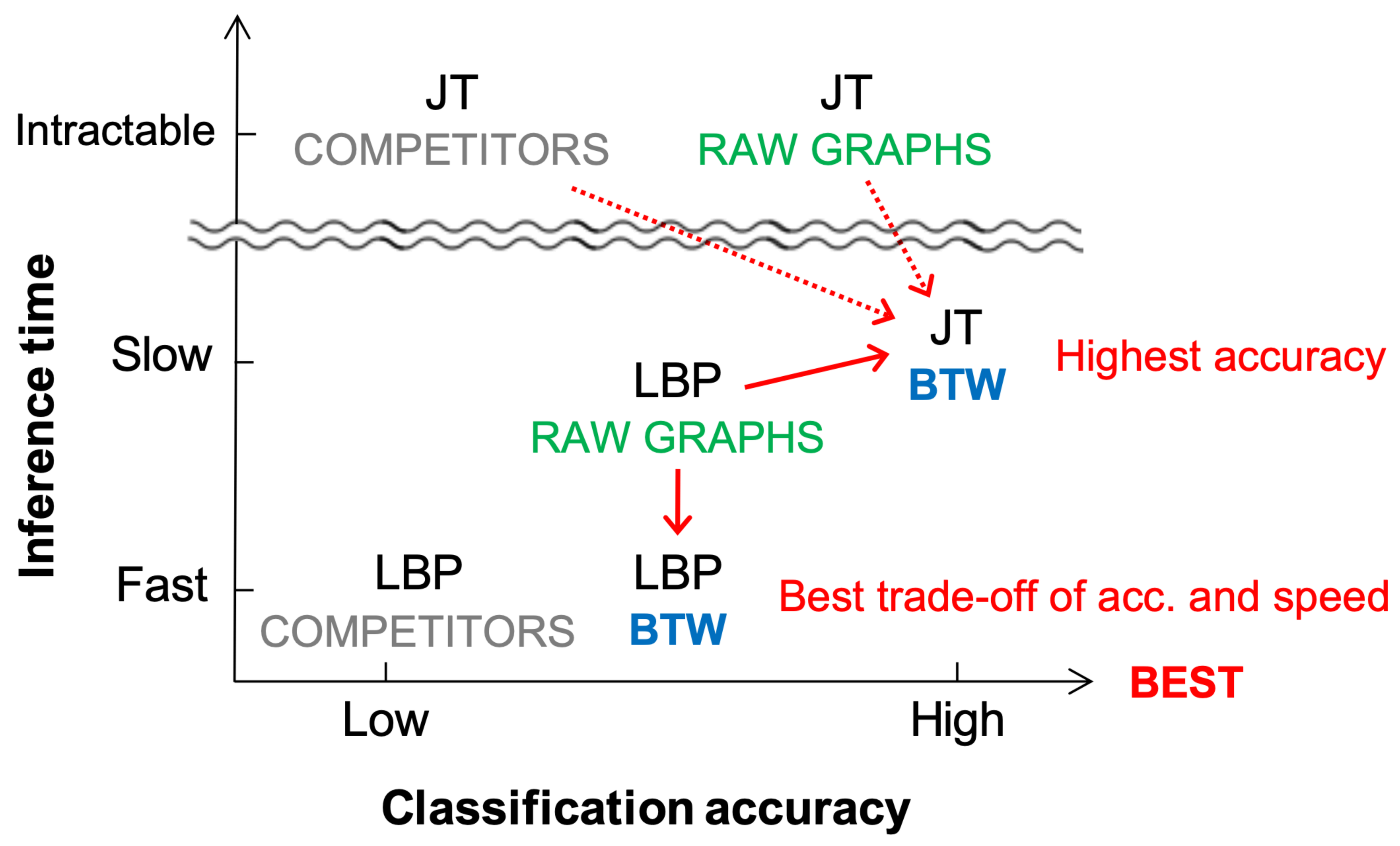
- *Can we sample subgraphs with bounded TW?*
  - **JT** on the subgraphs → **accurate** classification
  - **LBP** on the subgraphs → **fast** classification



PROPOSED APPROACH

Bounded Treewidth Sampling (BTW)

- Bounds the treewidth of subgraphs with k
- Maintains a k-tree K during a sampling process
- Use a score function m for sampling edges



Step 1: Initialization

- Given a graph G, BTW generate two graphs
  - A subgraph U_{k+1} having $k + 1$ nodes
  - A k-tree K_{k+1} which is a complete graph

Step 2: Incremental Updates

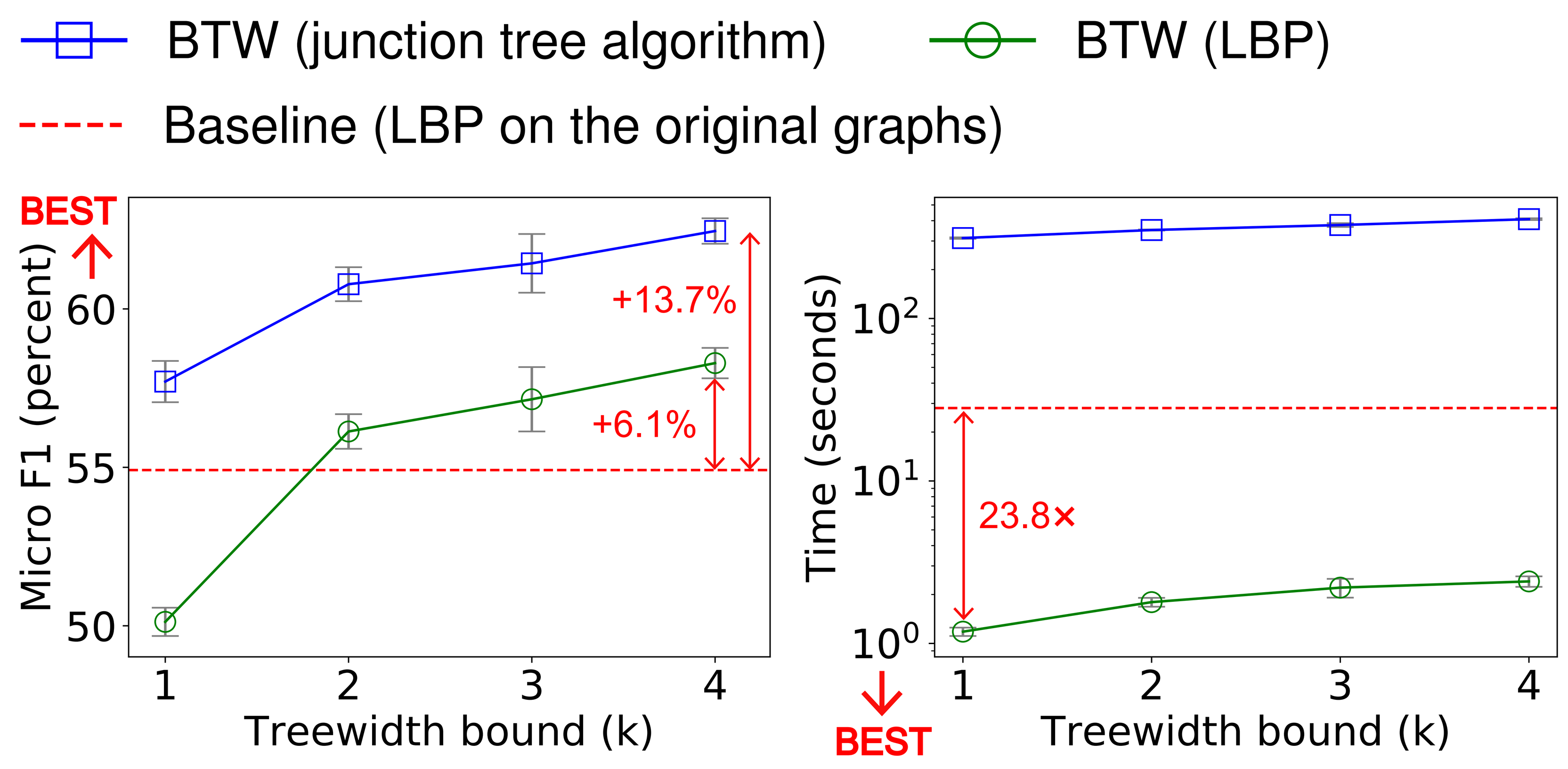
- Define a score function $m(u, C_k)$
  - u is a new node, and C_k is a k-clique in K_{k+1}
- Select $(u_{k+2}, C_k^*) = \text{argmax}_{u, C_k}\, m(u, C_k)$
- Connect u_{k+2} to C_k^* in both U_{k+1} and K_{k+1}

EXPERIMENTS

Experimental Setup

- Sample subgraphs using BTW (or others)
- Divide the labels for the k-fold validation
- Run JT or LBP for node classification
- **Datasets:** 4 real graphs with 2 – 16 labels
- **Evaluation:** classification accuracy & time

Effects for Node Classification



Comparison with Other Algorithms (by LBP)

Method	Wikipedia	CoRA	PubMed	PolBlogs
RE	35.3 ± 0.2	57.9 ± 0.3	61.8 ± 0.4	75.8 ± 1.0
RNE	51.3 ± 0.3	65.2 ± 0.2	71.1 ± 0.1	84.4 ± 0.6
HYB	49.4 ± 0.2	64.5 ± 0.2	69.8 ± 0.3	83.4 ± 0.4
RW	26.5 ± 2.7	43.0 ± 1.7	56.5 ± 1.2	65.2 ± 3.4
RJ	36.6 ± 0.4	55.4 ± 0.3	63.2 ± 0.4	75.6 ± 0.5
FS	29.8 ± 0.2	47.9 ± 0.2	56.2 ± 0.5	72.4 ± 0.8
FF	49.4 ± 0.2	63.7 ± 0.3	62.8 ± 0.4	79.8 ± 0.9
BTW	**56.1 ± 0.5**	**68.6 ± 0.3**	**74.8 ± 0.4**	**86.6 ± 0.9**