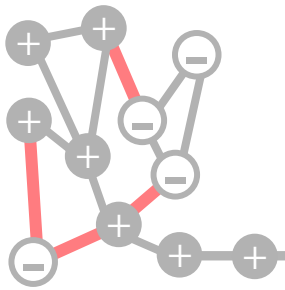


See the Tree through the Lines: The Shazoo Algorithm

T82

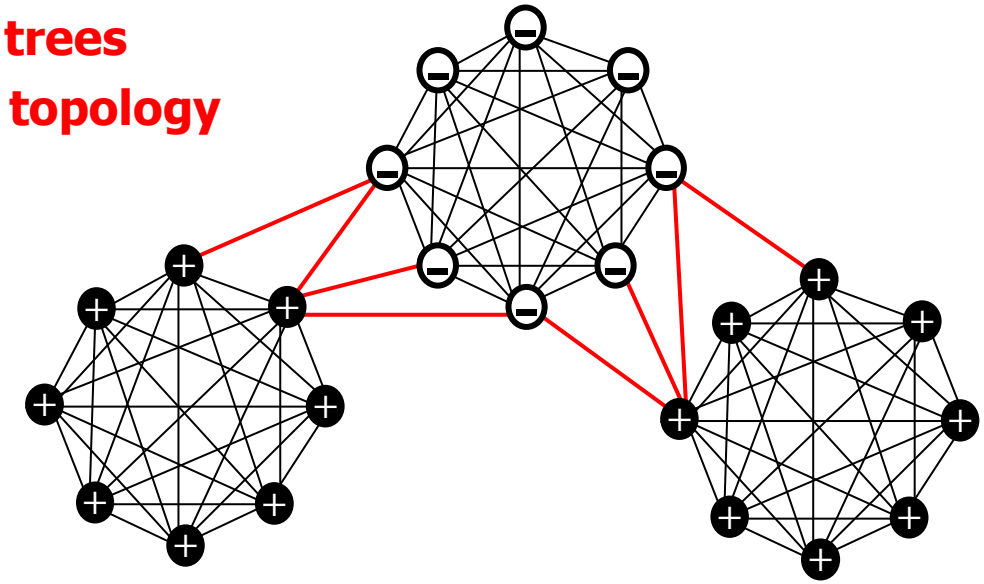
F. Vitale*, N. Cesa-Bianchi*, C. Gentile°, G. Zappella*

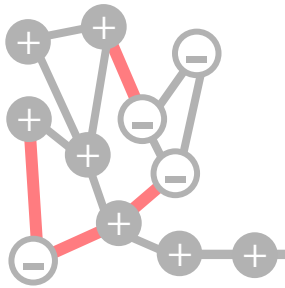
*Università degli Studi di Milano °Università degli Studi dell'Insubria

- **Learning on graphs/trees domains**: hyperlinked webpages, social networks, co-author networks, biological networks, …
- **Our learning problem: node classification** of weighted, connected and undirected **trees (and graphs)** based only on **graph topology**
- We focus on **binary labeling**
- **Bias: strongly connected nodes → same label**

   **= cut edge**

  weight of cut-edges is **small**





Learning protocol
The Shazoo algorithm

On-line learning protocol: Vertices are issued one by one in an arbitrary order $v_1, v_2, \ldots, v_n$

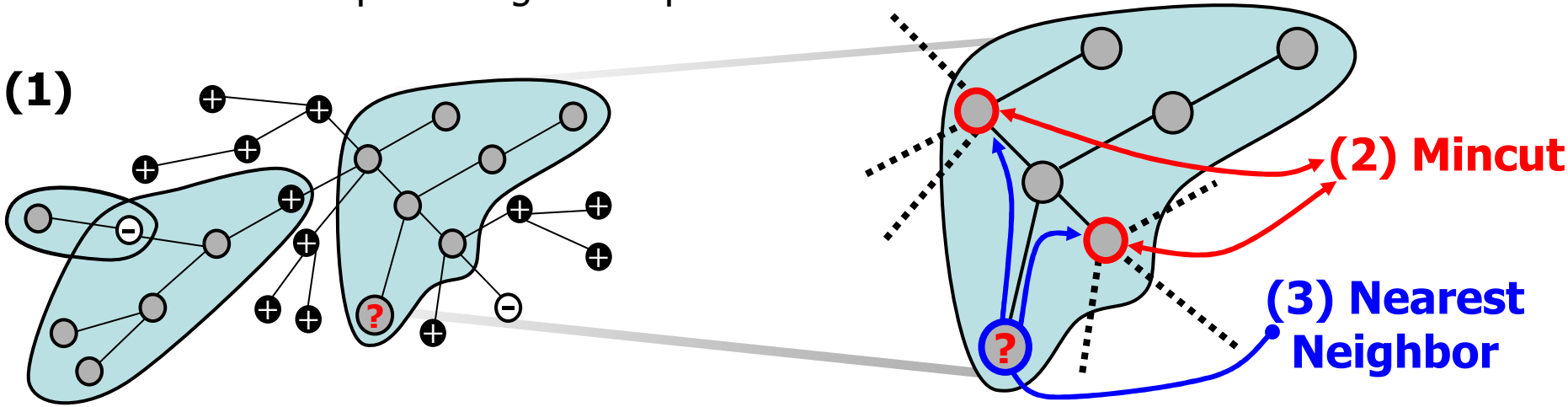
At each time step t: - learner **predicts** the label of v_t

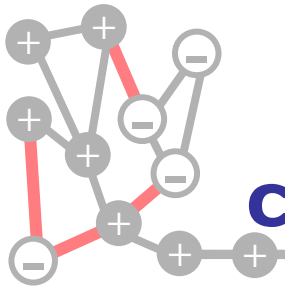
- learner **observes** the label of v_t

Goal: few prediction mistakes

- **The Shazoo algorithm: input = weighted trees T**
  (if the input is a graph G we can run Shazoo on a **spanning tree** T of G)

- **Shazoo (1) partitions** T into components (satisfying some properties), **(2)** uses **mincut** for estimating the labels of the component **border** vertices, **(3)** uses a **NN method** for predicting the required label





Analysis, implementation computational complexity and experiments

T82

Accuracy: **mistake bound** of Shazoo is **optimal** (up to log factors)

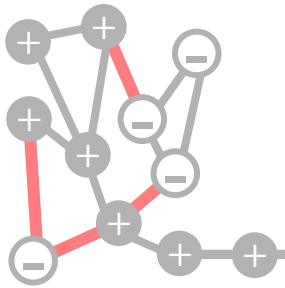
Implementation: simple and fast recursive method (based on sum-product algorithm) for using the mincut strategy

Time complexity:

- **On line protocol:** Worst case time per prediction: ***O*(#vertices)**
- **Batch protocol** (vertices are split into training and test sets) :
  Worst case time for predicting **all** labels of the test set: ***O*(#vertices)**

Space complexity: Linear in #vertices

Experimental results: Shazoo outperforms most of its competitors (e.g Label Propagation) on all our experiments on real-world datasets



Come to poster T82!

Shazoo

- Accuracy analysis: optimal mistake bound
- Scalability: very fast
- Easy to implement
- Works well in practice on real world datasets
- Easily extendible to multiclass prediction