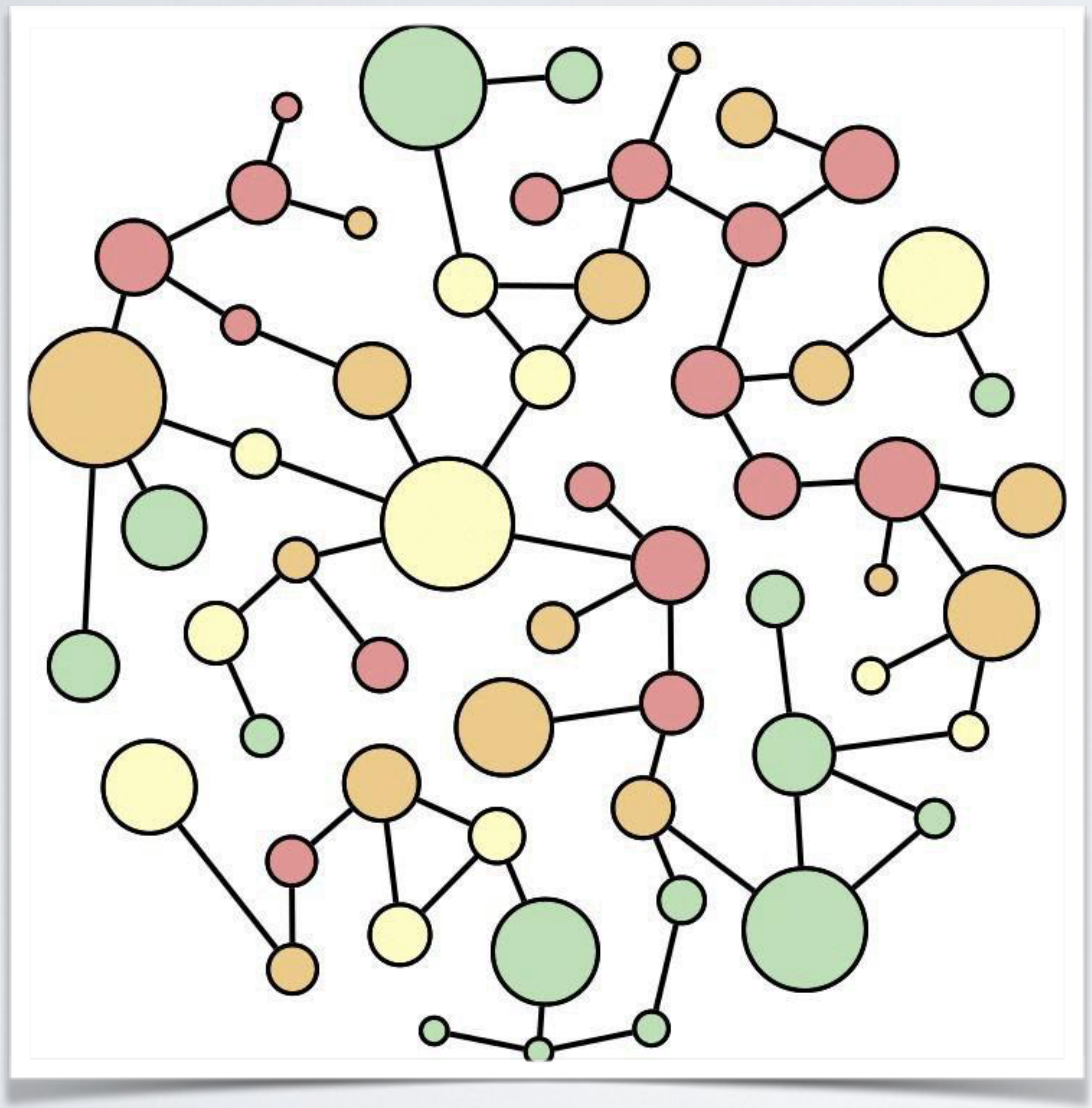
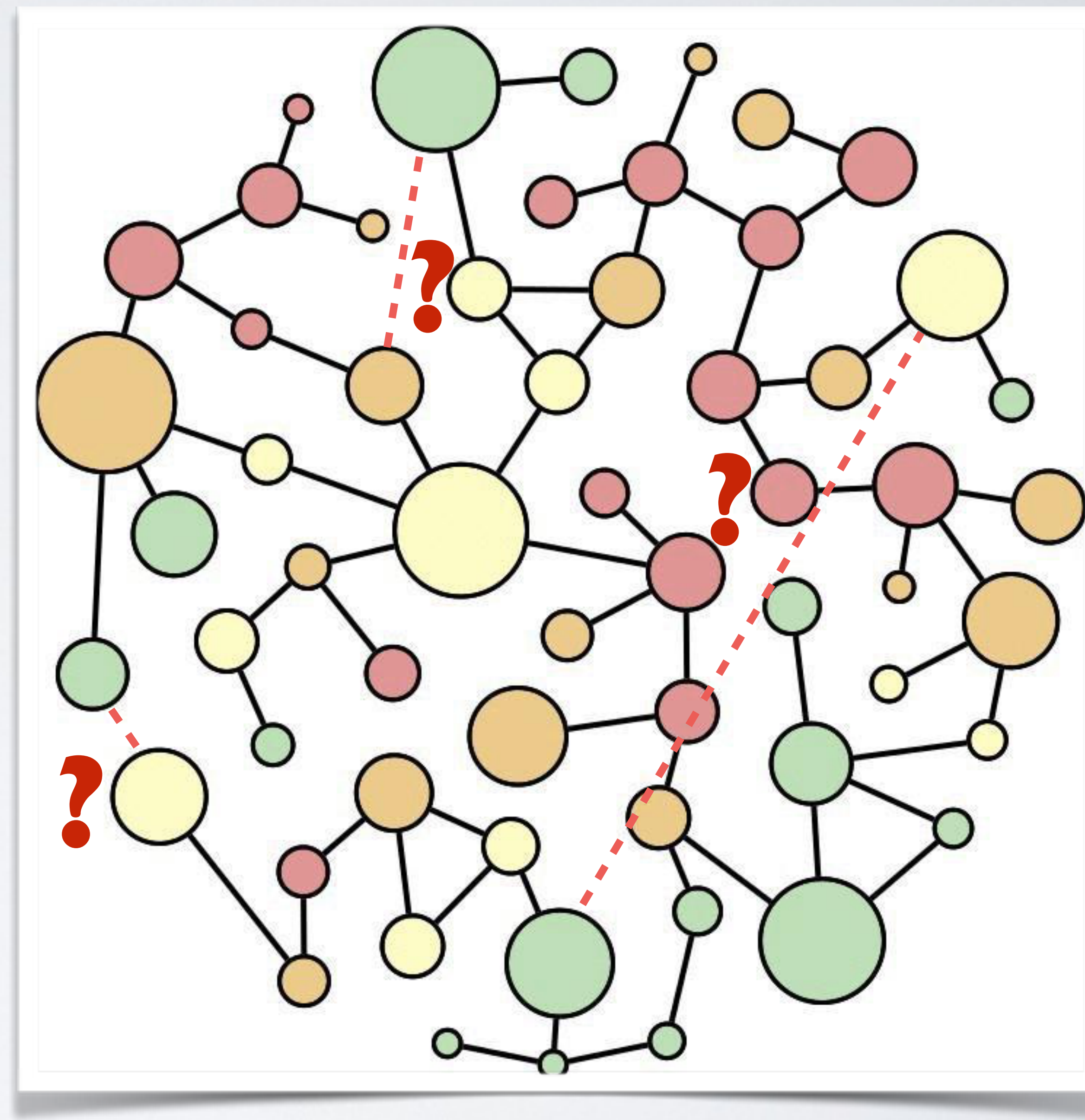
LINK PREDICTION USING GRAPH EMBEDDING

Aakash Sinha, Rémy Cazabet

LINK PREDICTION



Time T



Time T+1

LINK PREDICTION

- Numerous applications:
  - Recommender systems (Facebook, Spotify, Amazon,..)
  - Dynamic networks
  - Incomplete datasets
  - …

HEURISTICS

- Historically, link prediction is done using **heuristics**

- Example of heuristics:
  - The more neighbors in common between nodes, the highest chance of having an edge (CN)
  - The highest the degree of nodes, the highest the chance of having an edge (PA)
  - Many others (including more complex ones)

SUPERVISED LEARNING

- Heuristics give better results when combined using supervised learning (classifier)

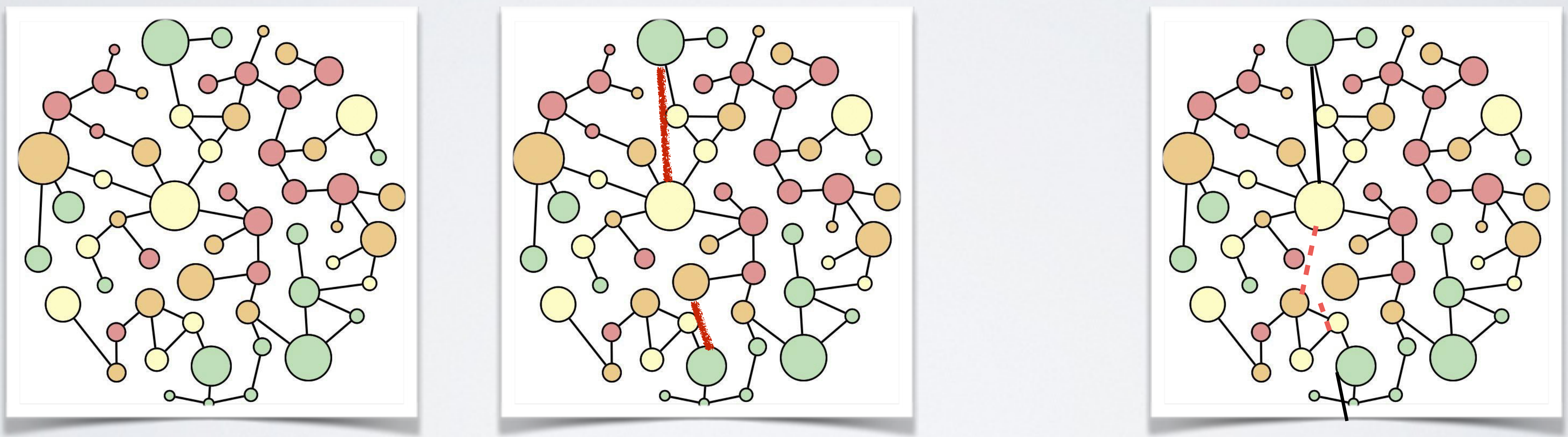
Classifier

Edge	CN	PA	AA	…	Label
(n1,n2)	0.3	0.2	0.34	…	Y
(n1,n3)	0.1	0.34	0.88	…	N
(n2,n3)	0.88	0.1	0.55	…	N
…	…	…	…	…	..

Edge	Prediction
(n1,n2)	0.88
(n1,n3)	0.5
(n2,n3)	0.2
…	…

SUPERVISED LEARNING

- Heuristics give better results when combined using supervised learning (classifier)



T | T+0.5 | T+1

Training | Prediction (testing)

USING EMBEDDINGS

- Embeddings provide a vector **by node**
- Generating one vector **by edge:**
  - Combine vectors of extremities
  - No theoretical arguments on how to combine
  - Best combine function decided empirically (best results)
    - Usually: Hadamar product

SUPERVISED LEARNING

Classifier

Edge	D1	D2	D3	…	Label
(n1,n2)	0.3	0.2	0.34	…	Y
(n1,n3)	0.1	0.34	0.88	…	N
(n2,n3)	0.88	0.1	0.55	…	N
…	…	…	…	…	..

Edge	Prediction
(n1,n2)	0.88
(n1,n3)	0.5
(n2,n3)	0.2
…	…

UNSUPERVISED LEARNING

- Embedding could also be used with **unsupervised** learning
- Distance between vectors in the embedding is **related to** the probability of having an edge between nodes
- =>The inverse of the distance between nodes in the embedding is the prediction

OUR QUESTION

- Previous articles have mostly focused on comparing graph embedding techniques between them
- Can we say that graph embeddings are (unambiguously) outperforming heuristics ?
  - If yes, by how much ?
  - If no, why and how to improve it ?

TESTING SET UP

- Methods (we should add more !)
  - Node2vec
  - VERSE
  - LE
  - HOPE

Heuristics	Definition
Common Neighbors	$\|\Gamma(u) \cap \Gamma(v)\|$
Adamic Adar	$\sum_{w \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\log \|\Gamma(w)\|}$
Preferential attachment	$\|\Gamma(u) * \Gamma(v)\|$
Jaccard Coefficient	$\frac{\|\Gamma(u) \cap \Gamma(v)\|}{\|\Gamma(u) \cup \Gamma(v)\|}$
Resource allocation index	$\sum_{w \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\|\Gamma(w)\|}$

- Graphs (we should add more !)
  - Facebook
  - AstropPH
  - VK
  - CoCit

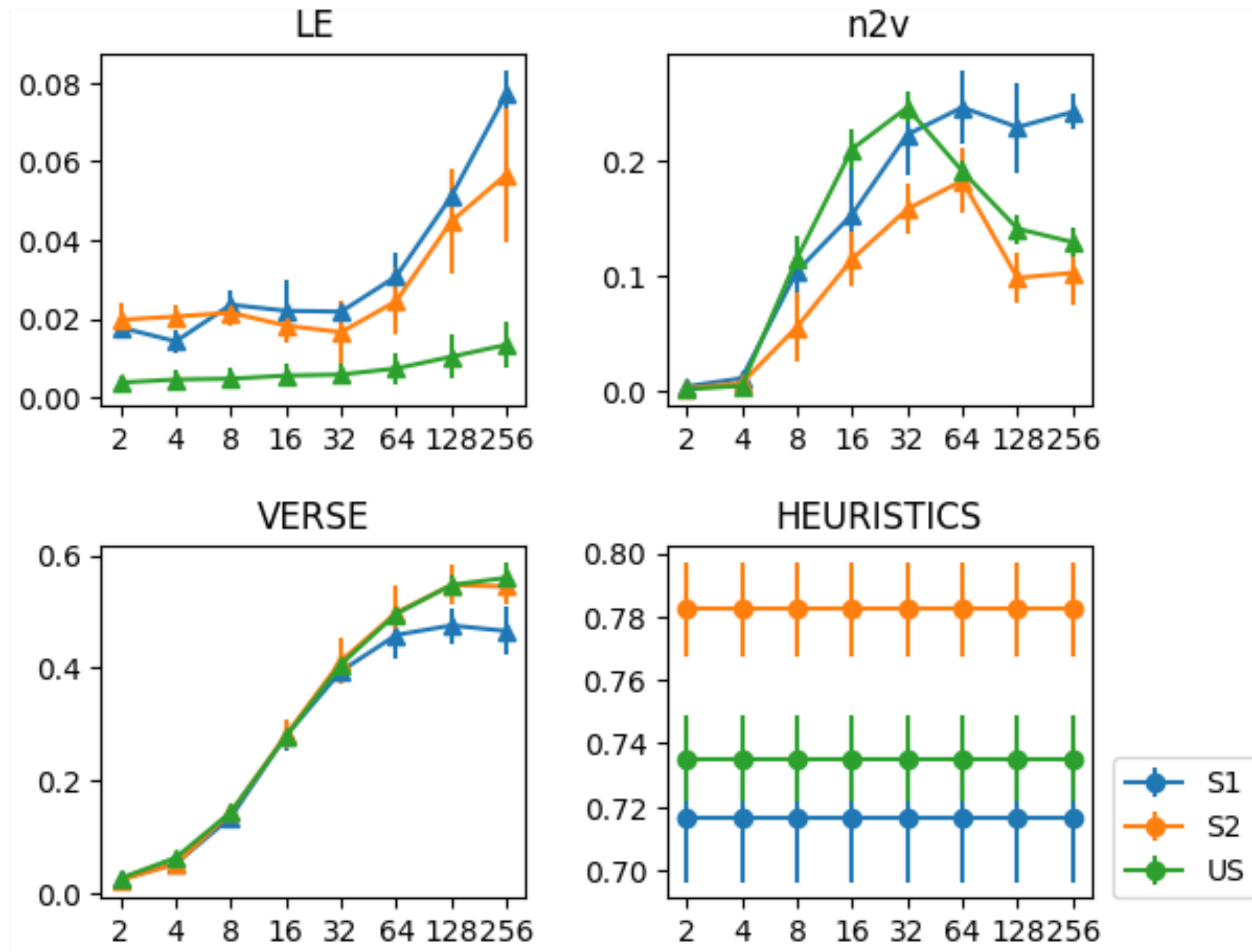
Name	|V|	|E|	Density
FACEBOOK [12]	4k	88k	0.0055
ASTROPH [11]	18k	198k	0.00061
VK [19]	79k	2.7M	0.00043

EVALUATION MEASURES

- Difficult choice. Link prediction has **high imbalance** between classes (density of real graphs is very low)
  - =>ROC score is independent from class distribution
  - =>AP is not but some authors prefer it (weights to the first few prediction)
  - =>Precision@k is not a single score, but easy to interpret.

- Chosen ones:
  - Average Precision (AP) (with a realistic unbalance)
  - ROC
  - Precision@k

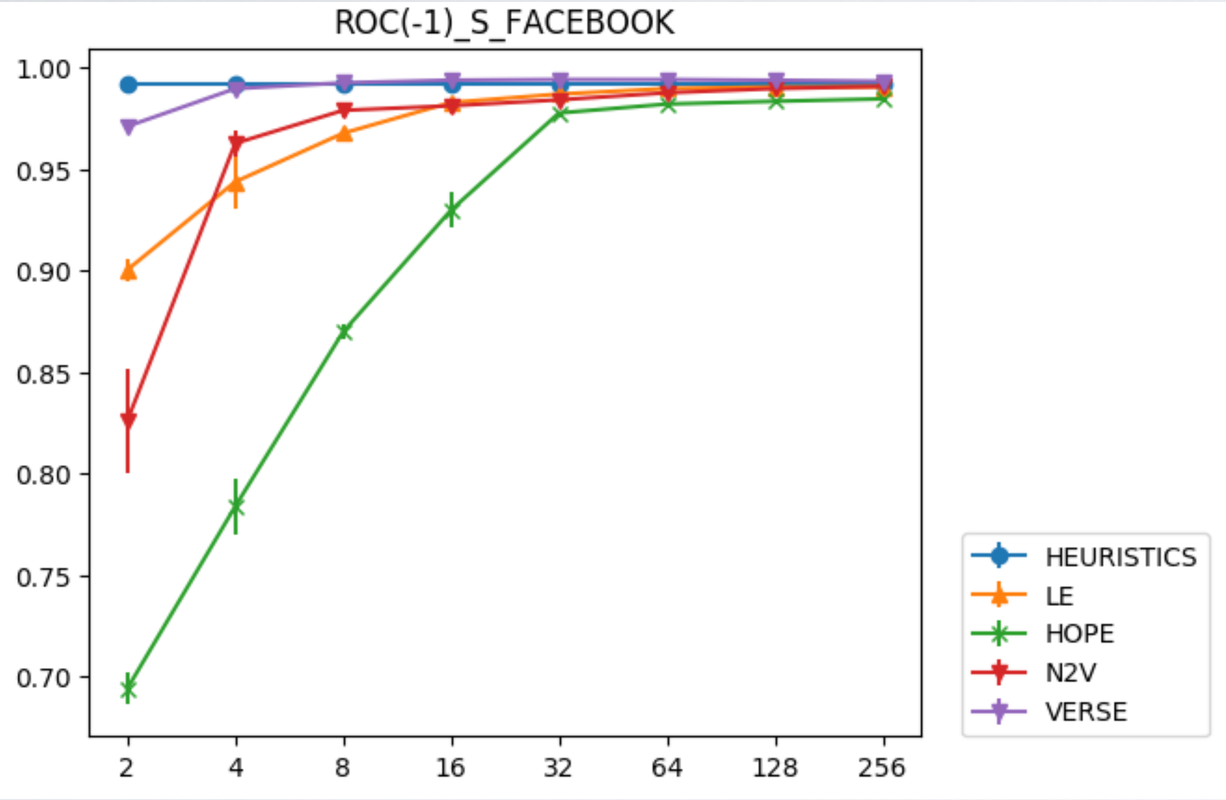
SUPERVISED OR UNSUPERVISED ?



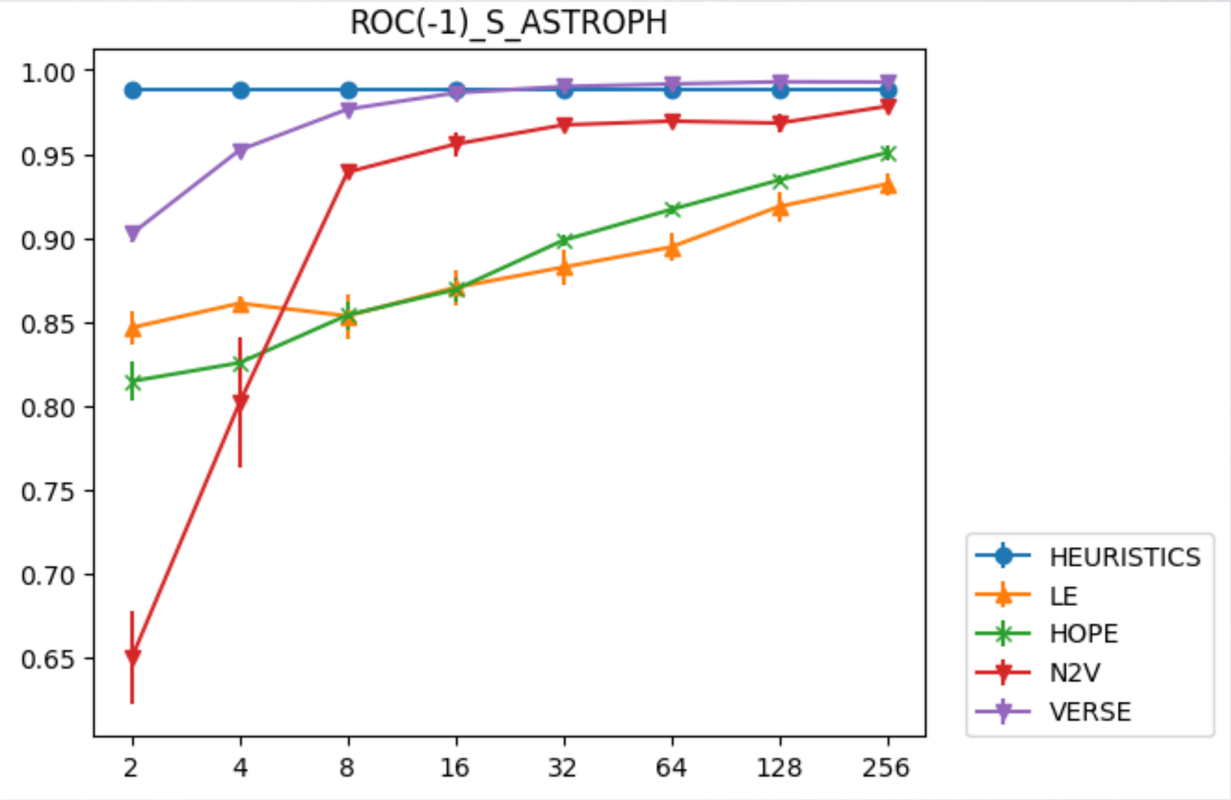
(b) ASTROPH

=> Supervised is usually more efficient than unsupervised (but not always that much)

WHICH APPROACH IS BEST ?



(a) FACEBOOK

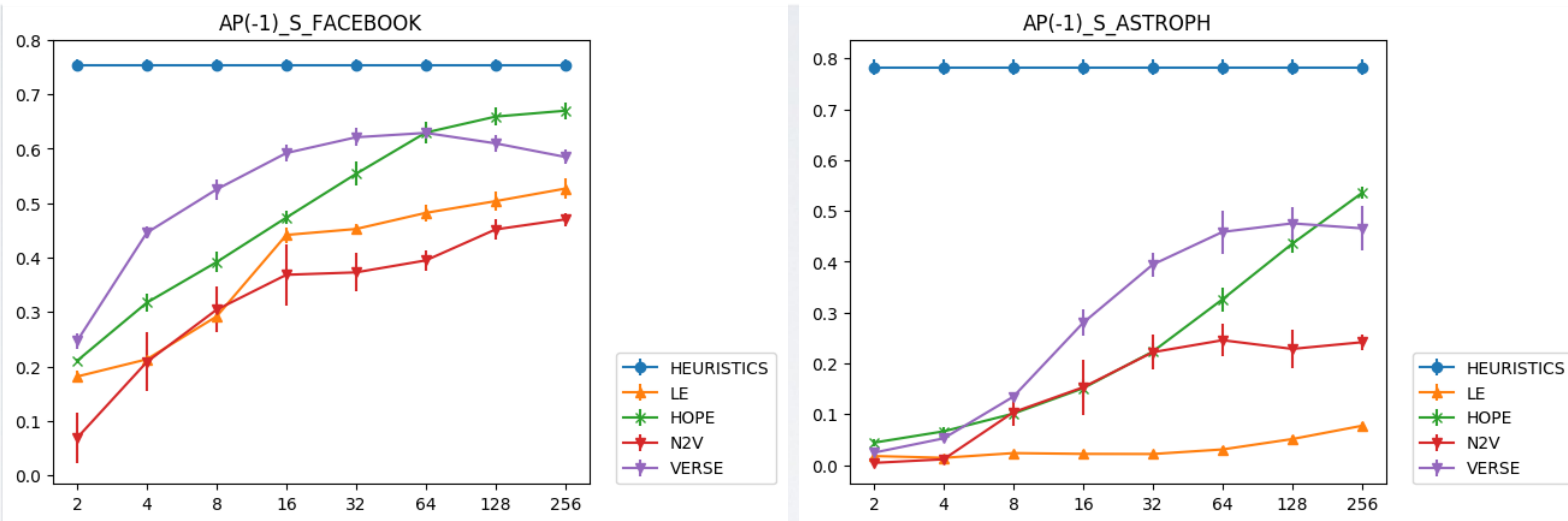


(b) ASTROPH

ROC Score

=>Only one embedding outperforms heuristics (VERSE)

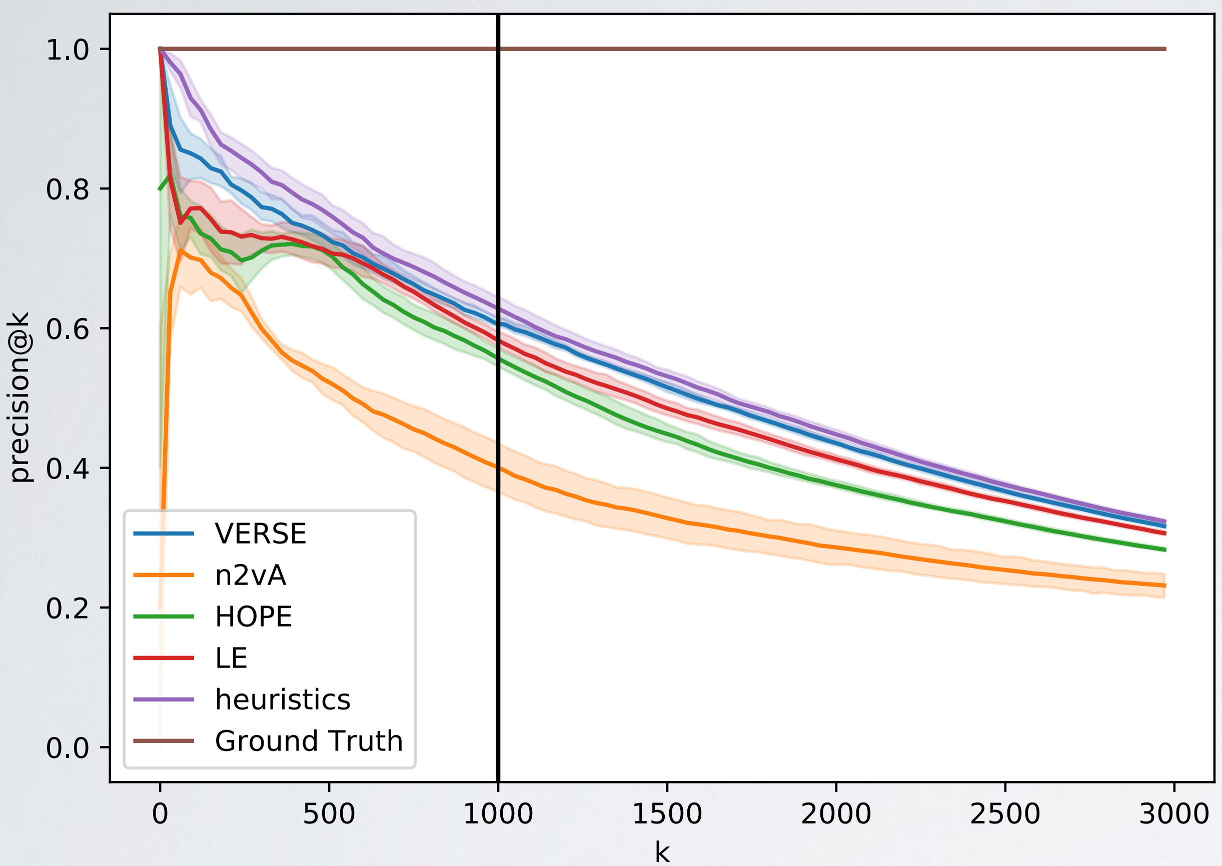
WHICH APPROACH IS BEST ?



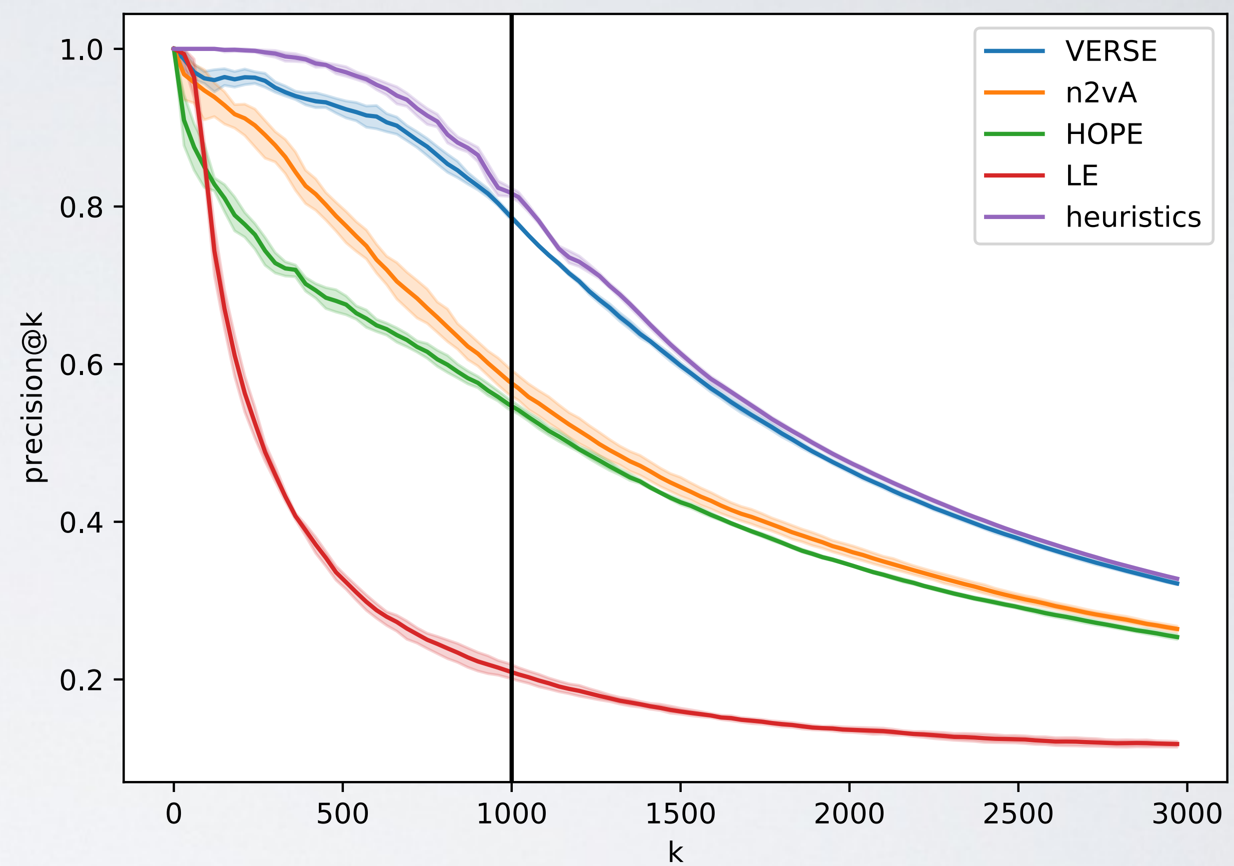
(a) FACEBOOK (b) ASTROPH

Average Precision (with a realistic
=> No embedding outperform heuristics

WHICH APPROACH IS BEST ?



FACEBOOK



ASTROPH

Precision @k

=>No embedding outperform heuristics

WHY ?

- Why embeddings do not outperform heuristics ???
  - (While they are much more advanced)
  - (And most published works seem to show the contrary)

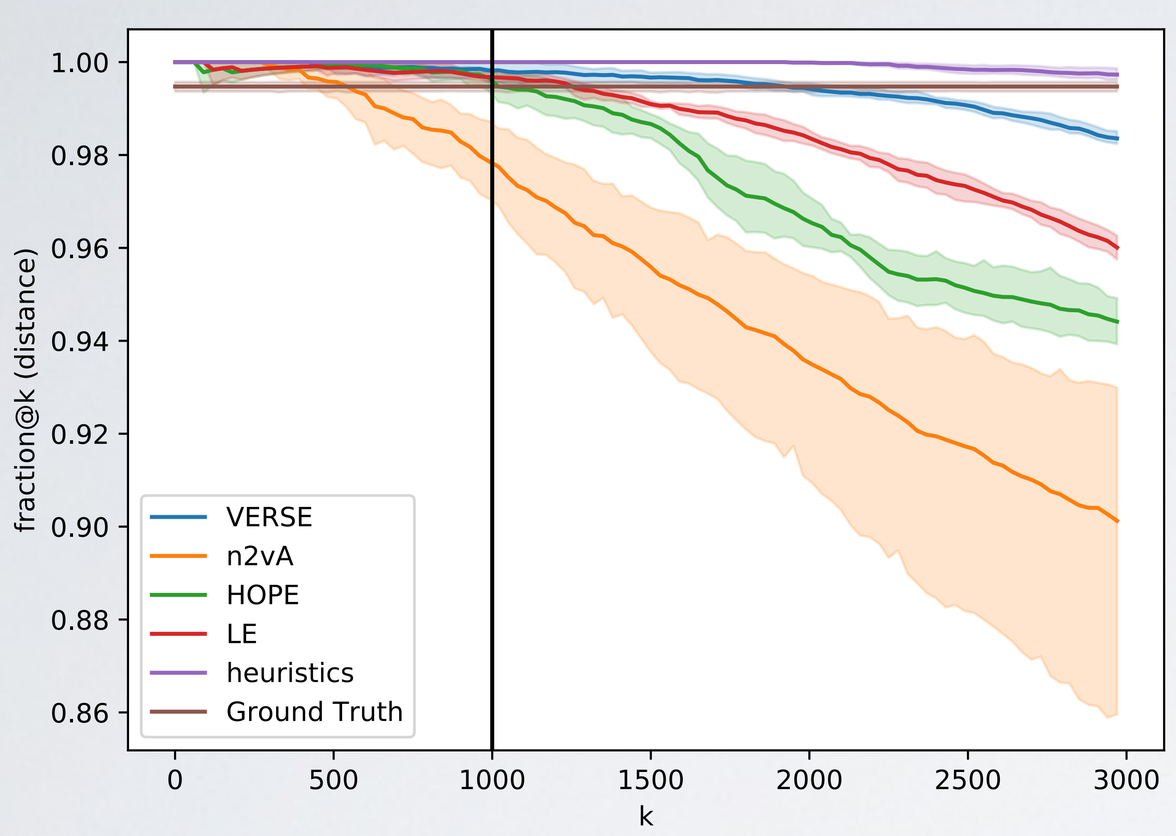
	Heuristics	Embeddings
AP	**0.13378**	0.02298
ROC	**0.813**	0.618

(a) Distance 2

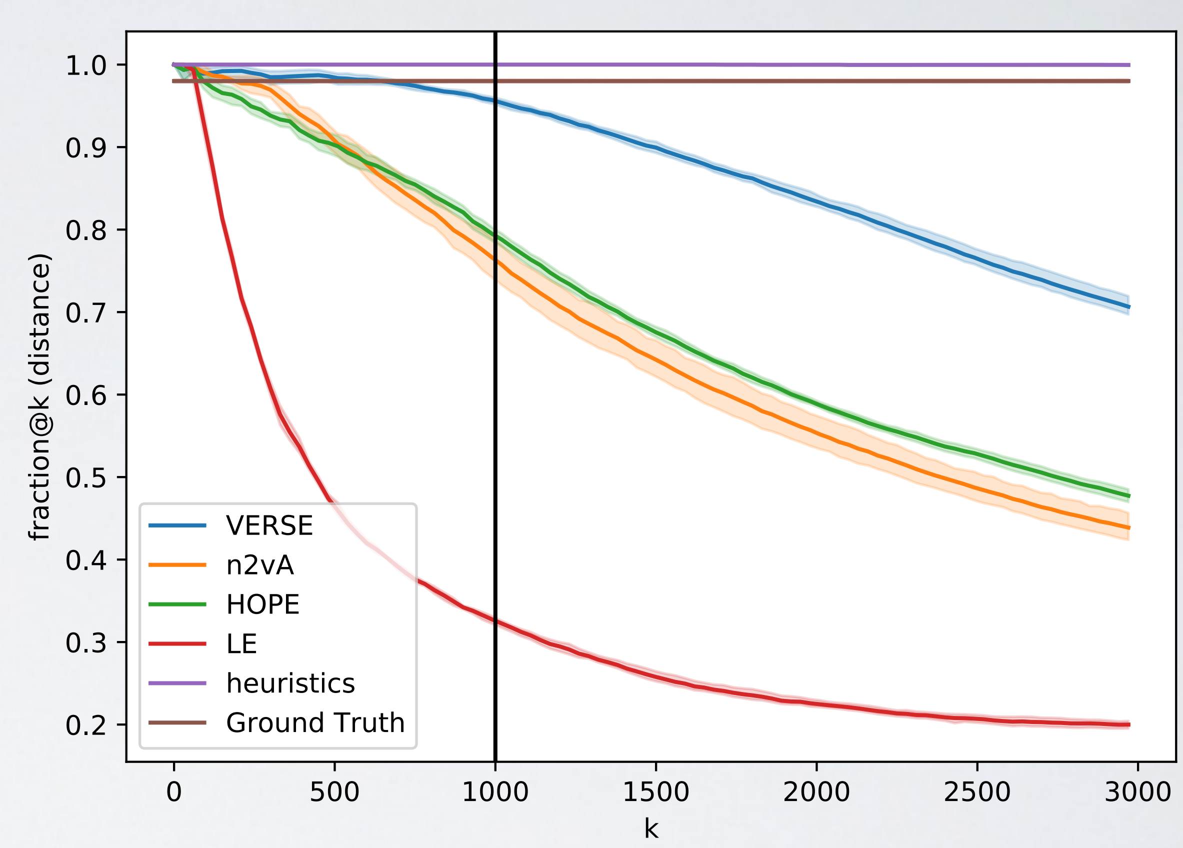
	Heuristics	Embeddings
AP	0.00219	**0.00338**
ROC	0.705	**0.794**

(b) Distance > 2

BIASES



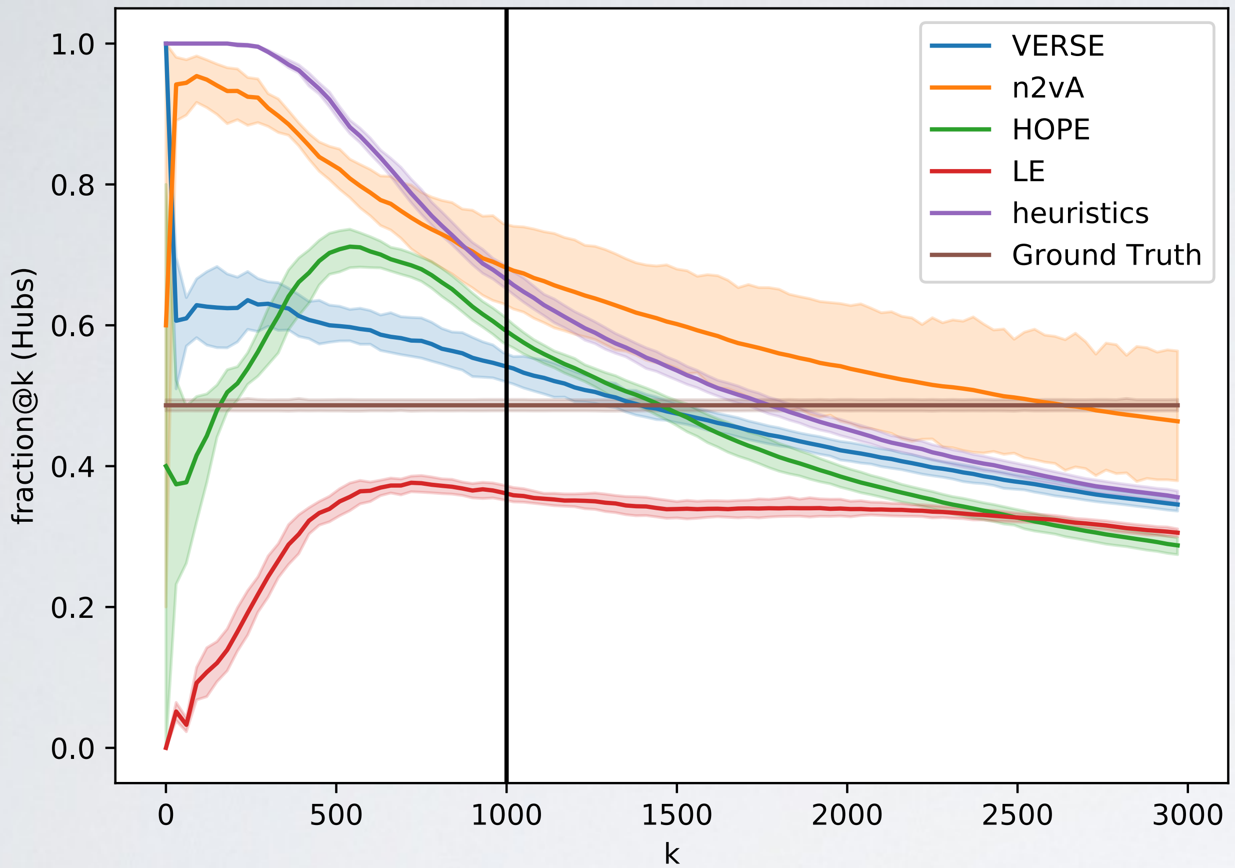
FACEBOOK



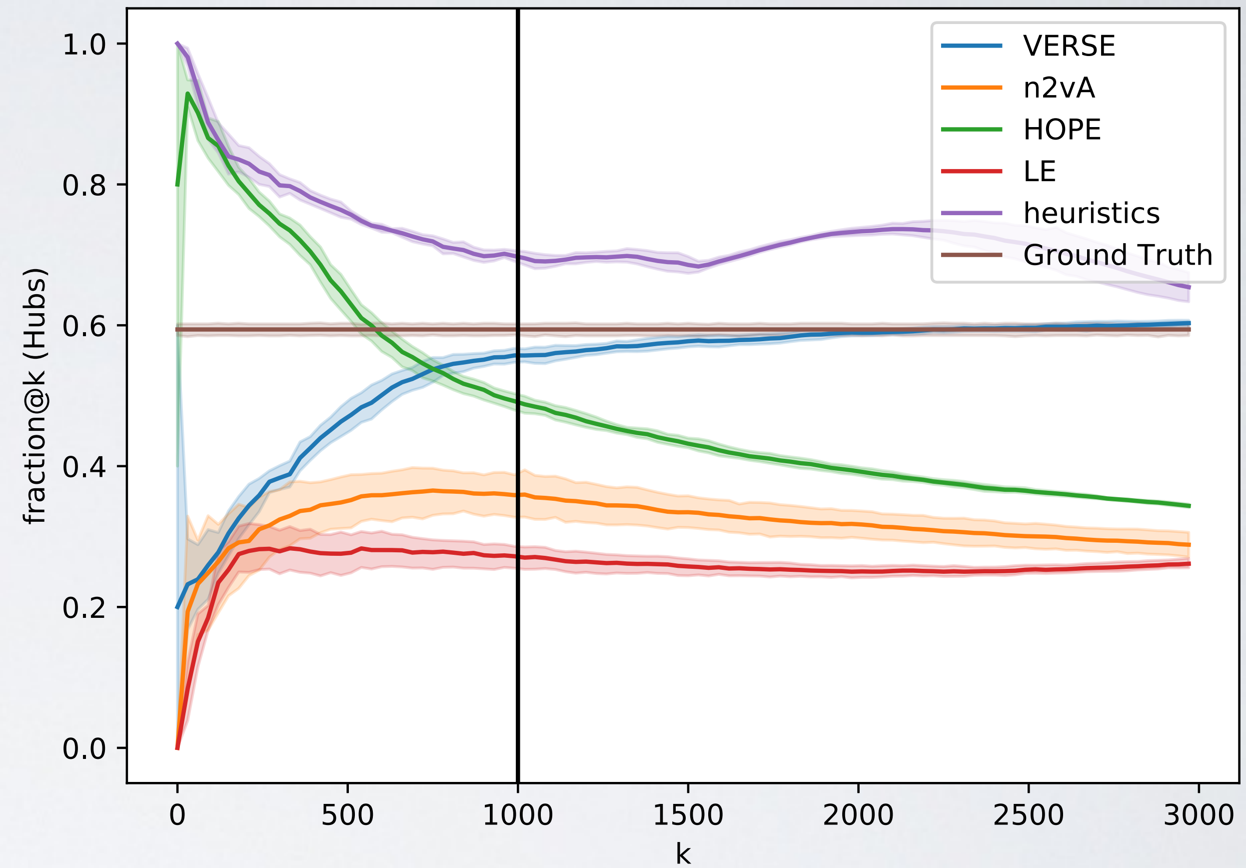
ASTROPH

Fraction@k of predictions at distance 2
=>Heuristics favor more the "easy" cases

BIASES



FACEBOOK



ASTROPH

Fraction@k of predictions including Hubs

BIASES

- Possible explanation (positive for embeddings):
  - => Embeddings try to predict “realistic” edges
  - => Heuristics focus only on the “simple” cases, the ones humans think should appear

  - => Heuristics results are more biased, which can be a problem
  - Social networks: recommend only people the most similar to you
  - Product/music recommendation: recommend only the most similar to your previous purchases
  - …

THANK YOU FOR YOUR ATTENTION

Comments and questions welcomed