ConGraT: Self-Supervised Contrastive Pretraining for Joint Graph and Text Embeddings


William Brannon


Wonjune Kang


Suyash Fulay


Hang Jiang


Brandon Roy

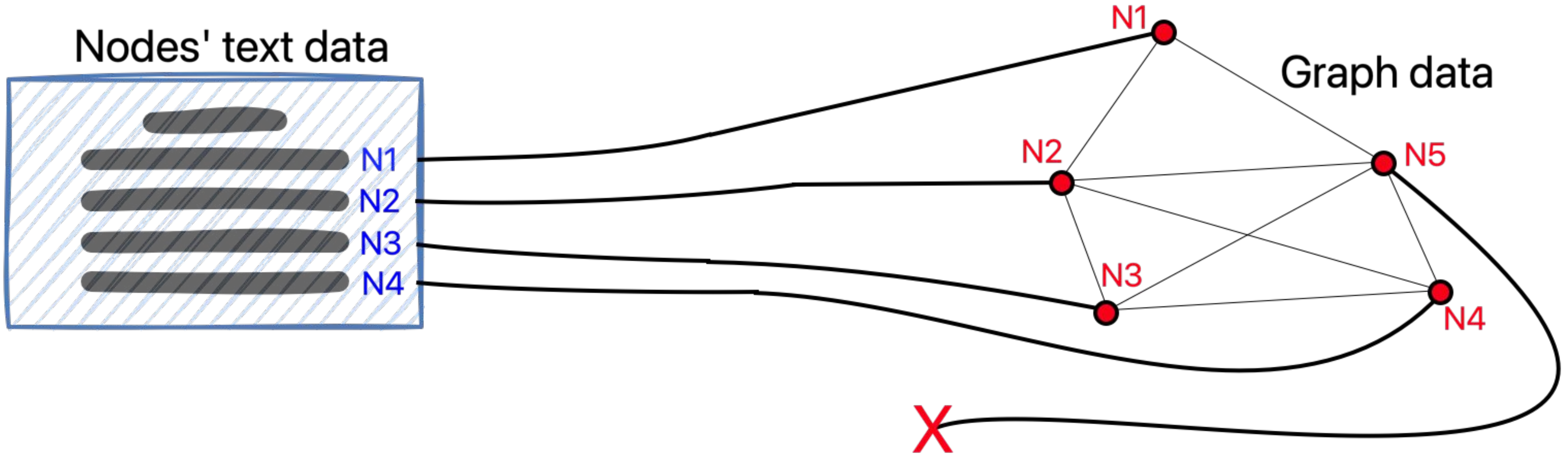

Deb Roy


Jad Kabbara

MIT Media Lab
MIT Center for Constructive Communication

TextGraphs-17

What’s the problem?



Text-attributed graphs

These crop up everywhere…

Social networks

Hyperlink graphs

Wikipedia
The Free Encyclopedia

Citations, news…

Google News

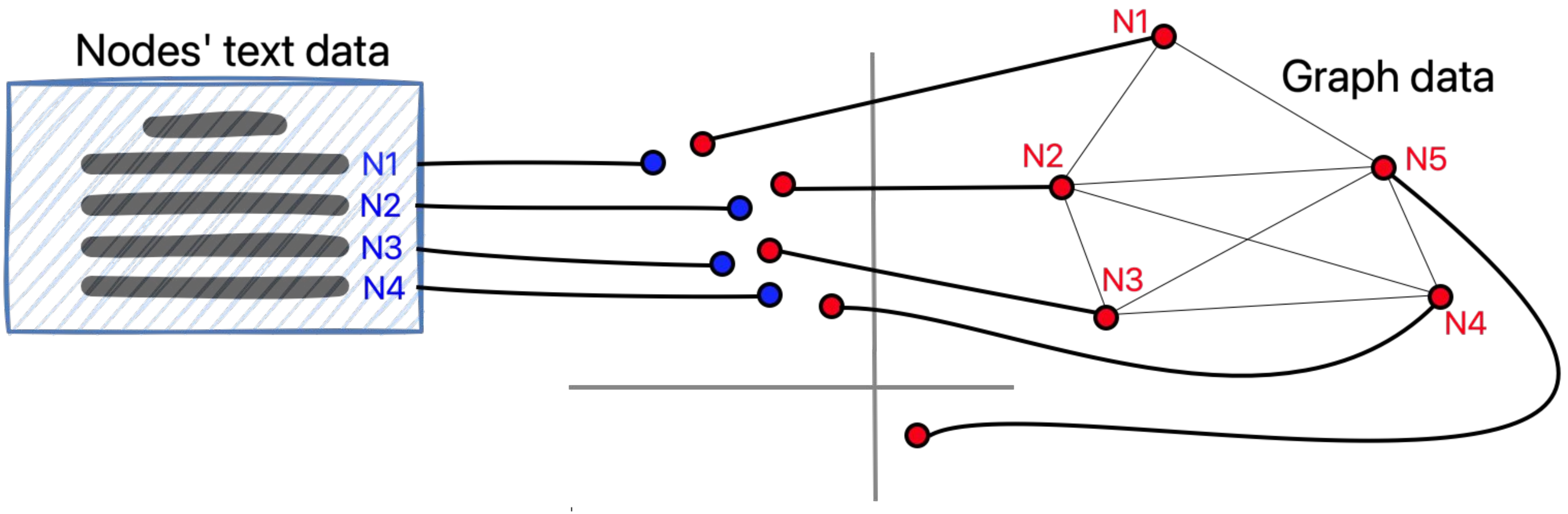
PubMed

…and relate to many applications

Recommendations: friends, products, related papers

Search: desired products, people, articles, …

Further afield, biology: gene/protein/etc interaction networks (we won’t talk about these further, though)



We want: joint embeddings

Text-augmented GNNs

- Feeding text info into a GNN somehow (Yang et al, 2015; Zhang et al, 2017)
- Don’t also produce text embeddings

Graph-augmented PLMs

- SPECTER (Cohan et al, 2020), LinkBERT (Yasunaga et al, 2022), SciNCL (Ostendorff et al, 2022)
- Don’t also produce node embeddings

Joint learning on TAGs

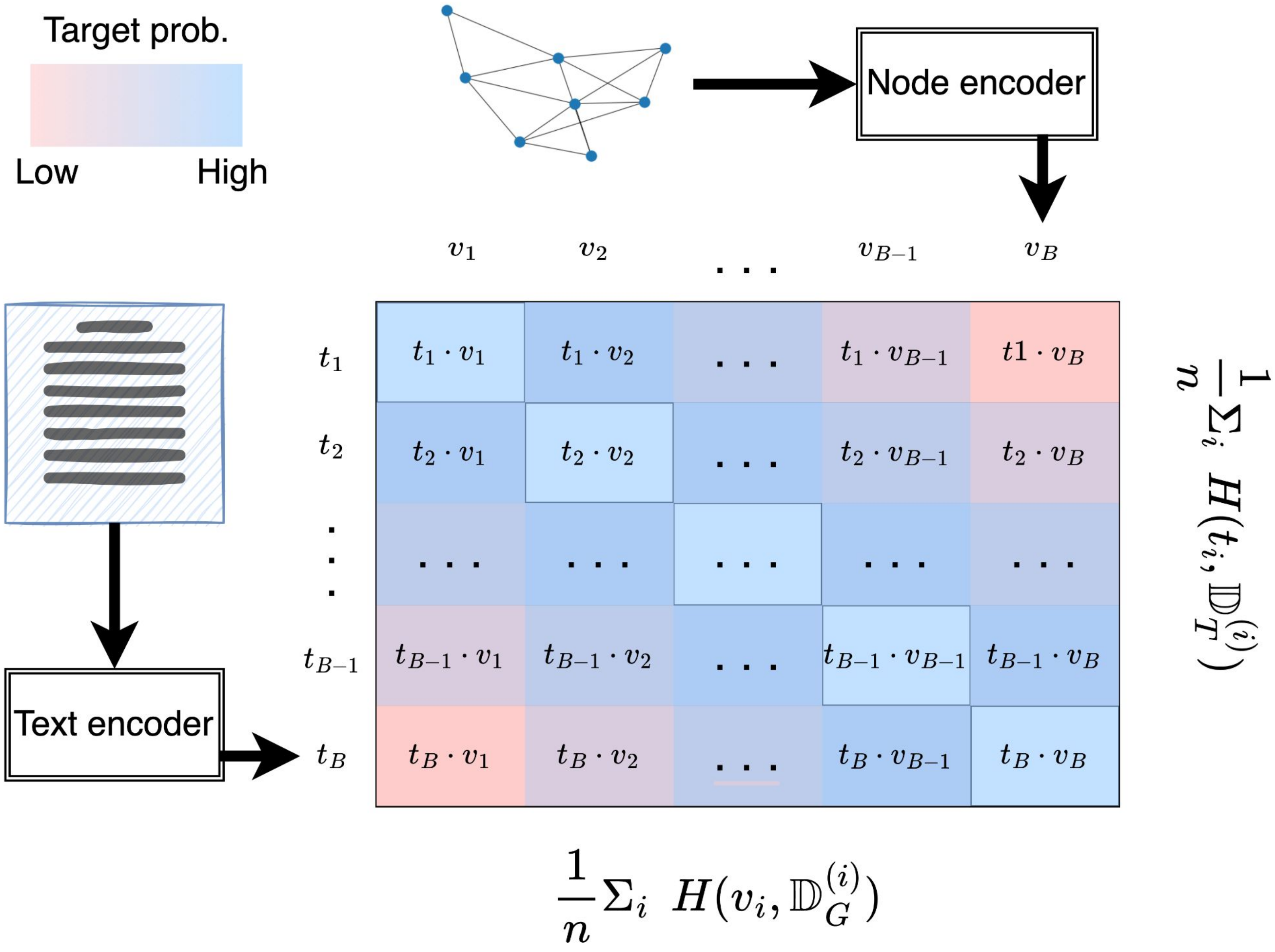
- GraphFormers (Yang et al, 2021), GIANT (Chien et al, 2023), GLEM (Zhao et al, 2022’)
- Usually pretty complex!

Related work

Model Architecture

High-level architecture

- Inspired by CLIP (Radford et al, 2021): we want a contrastive, self-supervised pretraining objective
- Within a minibatch, objective asks two questions:
  - Which node goes with which text?
  - Which text goes with which node?
- But! We incorporate graph-specific modifications:
  - It's hard to say how similar two images are, but graphs have lots of similarity measures
  - We "smooth" some of the probability mass across similar nodes and texts, not just the actual origin node/text
  - Similarity here is based on number of mutual neighbors two nodes share
- A theoretical interpretation: continuous relaxation of the CLIP objective across a node's n-hop (here, 2-hop) neighborhood, weighted by similarity
- **Our contribution**: this objective, applicable to a range of encoders



Target prob.
Low
High
Node encoder
Text encoder
v_1
v_2
v_{B-1}
v_B
t_1
t_2
t_{B-1}
t_B
$t_1 \cdot v_1$
$t_1 \cdot v_2$
$t_1 \cdot v_{B-1}$
$t1 \cdot v_B$
$t_2 \cdot v_1$
$t_2 \cdot v_2$
$t_2 \cdot v_{B-1}$
$t_2 \cdot v_B$
$t_{B-1} \cdot v_1$
$t_{B-1} \cdot v_2$
$t_{B-1} \cdot v_{B-1}$
$t_{B-1} \cdot v_B$
$t_B \cdot v_1$
$t_B \cdot v_2$
$t_B \cdot v_{B-1}$
$t_B \cdot v_B$
$\frac{1}{n}\Sigma_i \ H(t_i, \mathbb{D}_T^{(i)})$
$\frac{1}{n}\Sigma_i \ H(v_i, \mathbb{D}_G^{(i)})$

Datasets

About 8,700 politicians, journalists, entertainers; collected ca. 2021

Texts = tweets, edges = follow graph

Traditional graph benchmark of ~19k articles from Pubmed

Texts = articles, edges = cites

T-REx: ~9200 Wikipedia articles drawn from the T-REx dataset Elsahar et al (2018)

Texts = articles, edges = links

	Pubmed	T-REx	Twitter
# Nodes	19,716	9,214	8,721
# Edges	61,110	22,689	2,373,956
# Texts	59,381	18,422	167,558
# Classes	3	5	13 (5 tasks)

Dataset Statistics

Experiments

Setup

Experiments

- We want to show a general objective works across a general range of tasks
- Six ConGraT models per dataset (6, not 8, because directed edges only allow $\alpha = 0$):
  a. Text encoder: masked or causal/autoregressive
  b. Similarity info: $\alpha = 0$, $\alpha = 0.1$
  c. Edge directions: keep or discard?
- Text encoders:
  a. Masked: weights from sentence-transformers' all-mpnet-base-v2 (Song et al, 2020; Reimers and Gurevych, 2019)
  b. Causal: weights from DistilGPT2 (Sanh et al, 2019)
  c. Text-level representations by mean-pooling over the token representations
- Node encoders:
  a. Graph attention network or GAT (Veličković et al, 2017): 3 layers, 2 heads each, trained from scratch
- All embeddings are 768d; each dataset split into 70% train, 10% validation, 20% test

Baselines

Single-modality

Text: Transformer language model; MPNet or DistilGPT2 as appropriate to match the ConGraT model's initialization

Node: The same GAT as used in the ConGraT model, but trained with a graph autoencoding objective

Joint

LinkBERT: Uses network information to supervise language model training (Yasunaga et al, 2022)

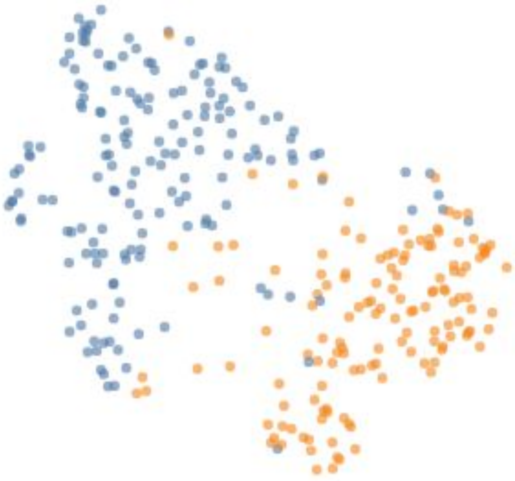
SocialLM: A lightly adapted version of SocialBERT (Karpov et al, 2021) which incorporates network info into vectors available to LM during training

Results

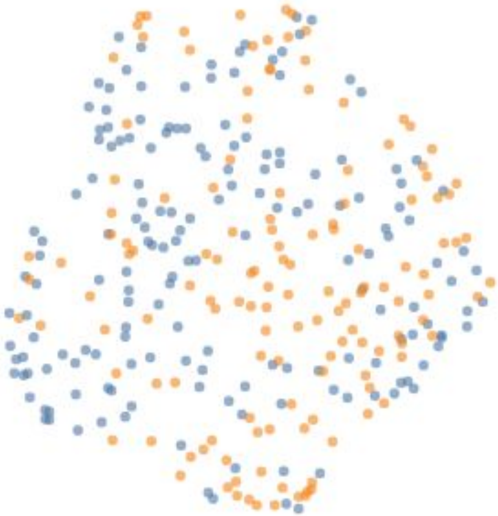
Visualize embedding geometry

US Congressional Twitter accounts:

- Color-coded by party: blue = D, orange = R
- Plots depict 2D UMAP ([McInnes et al, 2018](#)) of node embeddings from each model



ConGraT embeddings (with α = 0)



GAT baseline embeddings

How well can we predict classes of nodes from each model's text or graph embeddings?

Classes: Twitter

- Demographic variables for users from Wikipedia data
- Age, gender, race, geographic region, political party, occupation (politico/entertainer)

Classes: Pubmed

- Article subjects: Type I, Type II or experimental evidence

Classes: T-REx (Wikipedia)

- Top 5 Wikipedia article categories (see paper for category selection details)

Node Classification

		Age		Gender		Occupation		Party		Region	
		C	M	C	M	C	M	C	M	C	M
Graph	ConGraT ($\alpha = 0$)	0.646	0.665	**0.811**	**0.802**	**0.993**	0.989	**0.966**	0.959	**0.755**	**0.780**
	ConGraT ($\alpha = 0.1$)	**0.650**	**0.682**	0.803	0.801	0.992	**0.993**	0.960	**0.986**	0.742	0.774
	GAT	0.631	0.631	0.713	0.713	0.967	0.967	0.757	0.757	0.678	0.678
Text	ConGraT ($\alpha = 0$)	**0.622**	**0.628**	0.663	**0.668**	**0.961**	**0.959**	**0.771**	0.787	**0.693**	0.679
	ConGraT ($\alpha = 0.1$)	0.620	0.624	**0.668**	0.661	0.960	0.958	0.771	**0.796**	0.686	**0.680**
	LinkBERT	–	0.617	–	0.661	–	0.954	–	0.762	–	0.606
	Social-LM	0.566	0.567	0.602	0.608	0.921	0.909	0.628	0.676	0.582	0.572
	Unimodal LM	0.610	0.613	0.649	0.655	0.948	0.945	0.742	0.769	0.587	0.598

Node Classification: Twitter

AUC values from logistic regression

Predictors:

- *Graph*: Predict from node embedding
- *Text*: Predict from centroid of text embeddings

C = causal, M = masked

		Pubmed		T-REx	
		C	M	C	M
Graph	ConGraT ($\alpha = 0$)	0.967	**0.964**	**0.951**	0.937
	ConGraT ($\alpha = 0.1$)	**0.973**	0.963	0.949	**0.946**
	GAT	0.956	0.956	0.939	0.939
Text	ConGraT ($\alpha = 0$)	0.962	0.958	0.920	0.911
	ConGraT ($\alpha = 0.1$)	**0.969**	**0.966**	**0.931**	**0.928**
	LinkBERT	–	0.954	–	0.906
	Social-LM	0.858	0.878	0.890	0.851
	Unimodal LM	0.931	0.943	0.908	0.892

Node Classification: Pubmed + Wiki

How well can we predict edge existence between nodes?

We use inner product decoding (Kipf and Welling, 2016) to get predicted probabilities of edge existence

			Pubmed	T-REx	Twitter
Masked	$\alpha = 0$	U	0.953	0.899	0.791
		D	0.952	0.902	0.797
	$\alpha = 0.1$	U	**0.980**	**0.951**	**0.802**
Causal	$\alpha = 0$	U	0.956	0.908	**0.806**
		D	0.950	0.897	0.799
	$\alpha = 0.1$	U	**0.979**	**0.957**	0.799
GAT	–	U	0.943	0.927	0.713
		D	0.940	0.925	0.723

U = undirected, D = directed

Link Prediction

Does pretraining with a ConGraT objective improve LM performance?

Yes! Perplexity is lower if the LM is first pretrained with a ConGraT objective before fine-tuning on training-set text.

	Pubmed	T-REx	Twitter
$\alpha = 0$	6.95	**15.99**	16.08
$\alpha = 0.1$	**6.94**	16.07	**16.07**
LM	6.98	16.84	16.44

Figures are test-set perplexity

Language Modeling, vs unimodal LM

Application:
Community Detection

What do we want to do?

Can we detect communities informed by not just network structure, but also text?

Experiment setup:

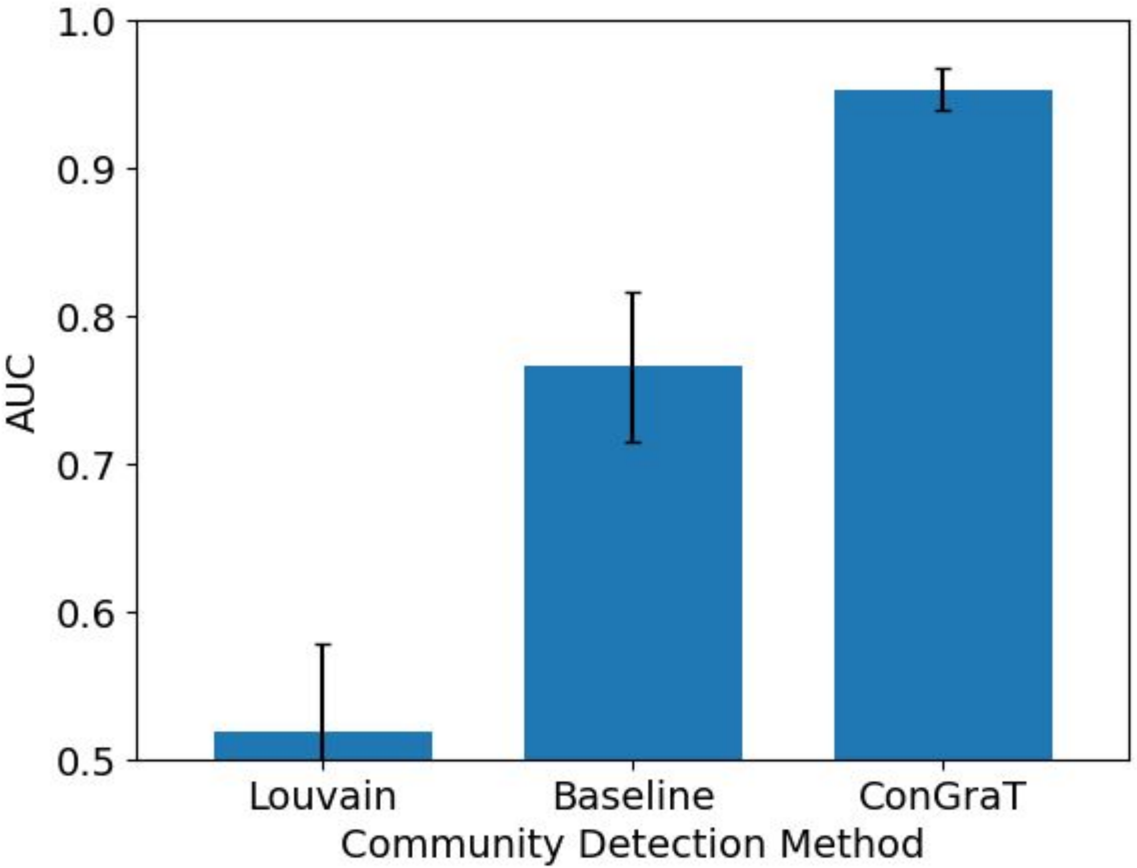
1. Generate three sets of community labels:
   a. Louvain algorithm, baseline (Blondel et al, 2008)
   b. Cluster GAT baseline node embeddings using UMAP (McInnes et al, 2018) and HDBSCAN (McInnes et al, 2017)
   c. Cluster ConGraT node embeddings using same method
2. For each label set, for each user, predict community label from the user's **text** embeddings

Q: Is community membership more predictable from text using ConGraT embeddings?

Yes!

Our method produces much more textually informed communities than baselines!

(Plots show AUC on prediction task.)



Thank you!

Questions, comments, want to collaborate? Get in touch!

wbrannon@mit.edu

arxiv.org/abs/2305.14321

Check out the paper!

aclanthology.org/2024.textgraphs-1.2/

Paper

github.com/wwbrannon/congrat

Code

Sensitivity Analysis

	NCG	NCT	LP	LM
$\alpha = 0.0$	0.967	0.962	0.956	6.95
$\alpha = 0.1$	**0.973**	**0.969**	**0.979**	6.94
$\alpha = 0.5$	0.962	0.958	0.977	6.98
$\alpha = 1.0$	0.941	0.900	0.897	**6.88**
Baseline	0.956	0.931	0.943	6.98

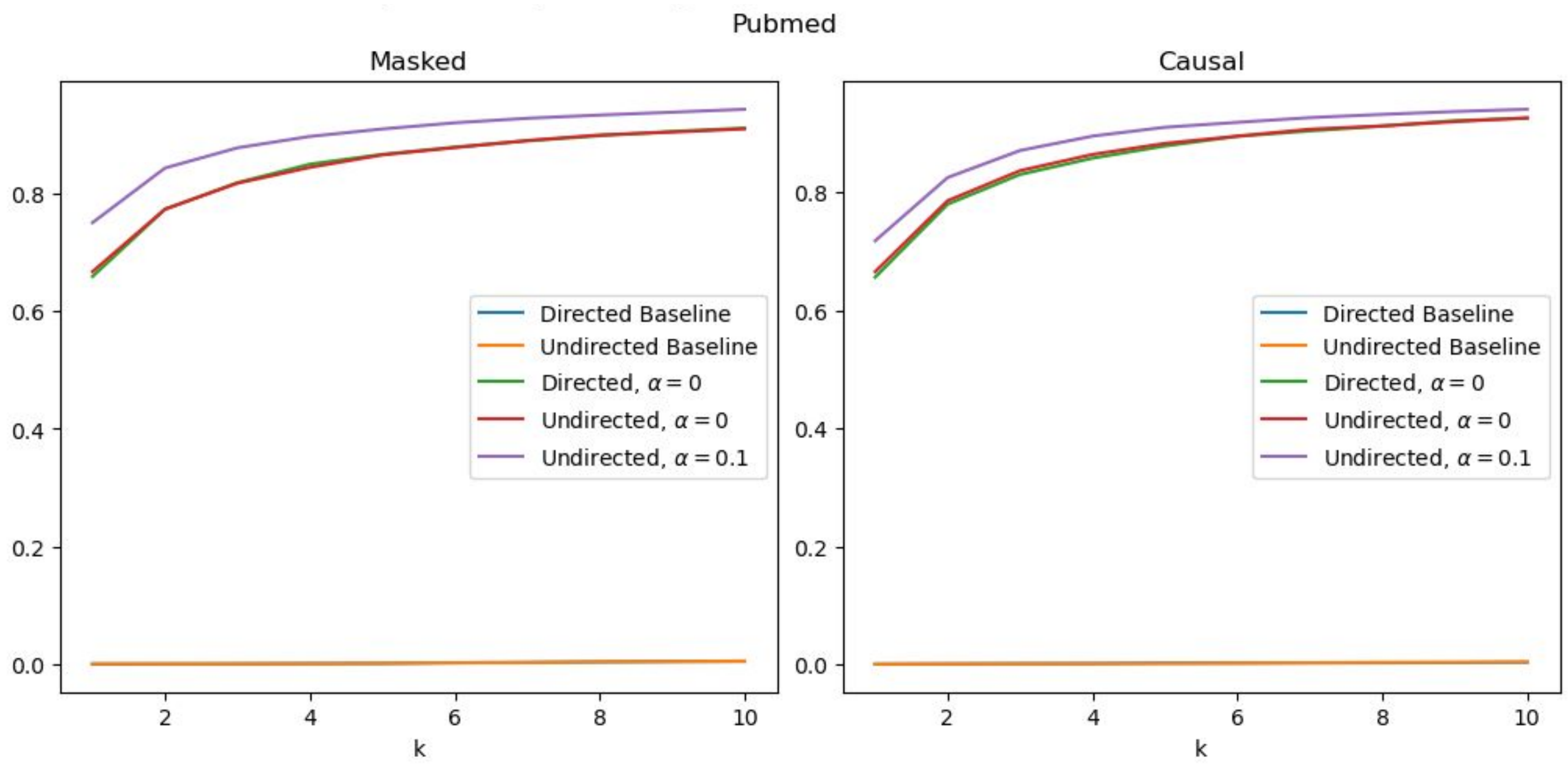
Table 6: Results of sensitivity analysis. NCG = node classification, graph; NCT = node classification, text; LP = link prediction; LM = language modeling. Values are AUC for the first three columns and perplexity for language modeling.

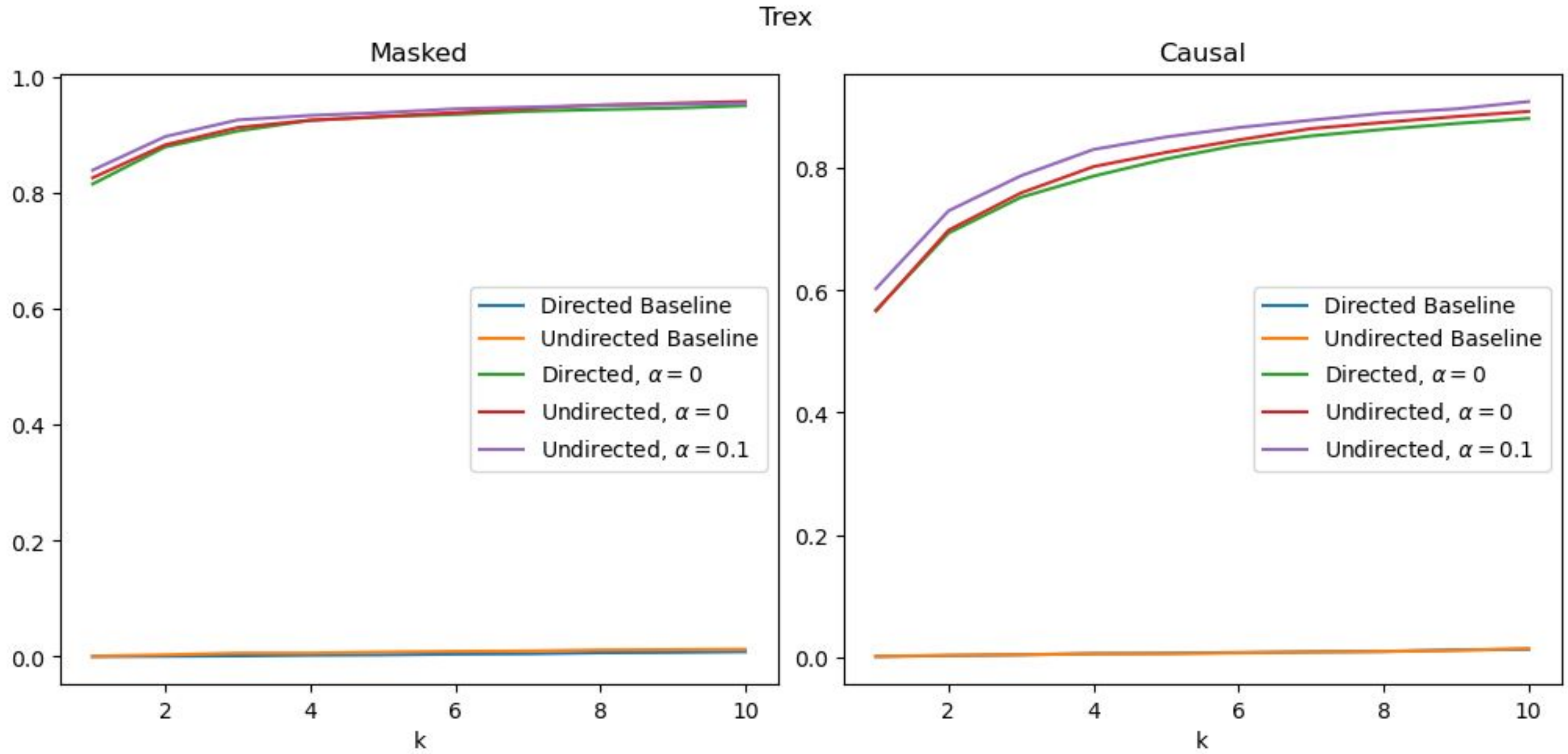
Embedding Space Geometry Analysis

Distance Correlation

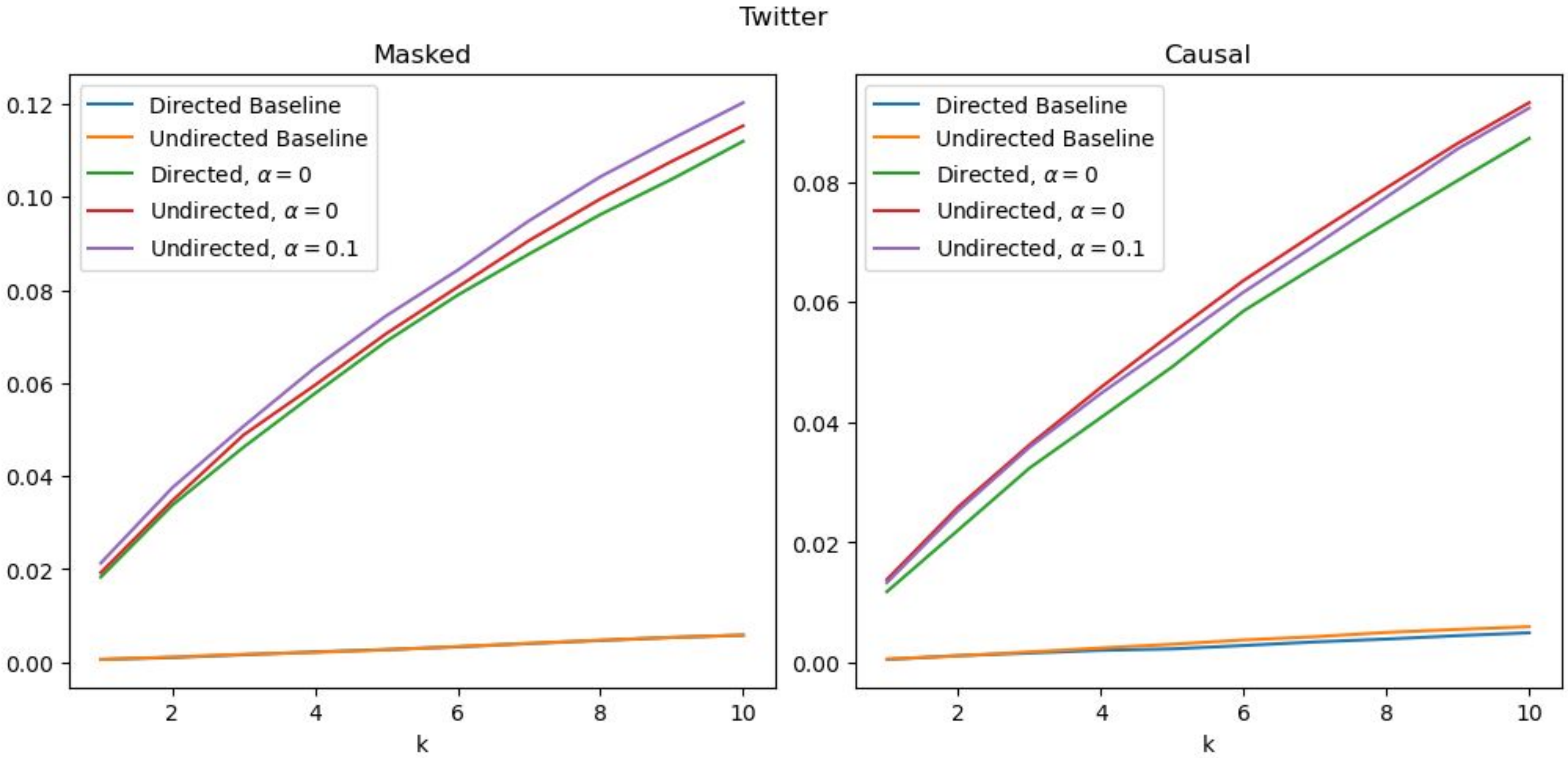
Dataset	Directed	LM Type	Sim.	Inter-Embedding		Text Emb.-Graph	
				Joint	Separate	Joint	Separate
Pubmed	Directed	Causal	$\alpha = 0.0$	**0.682**	0.100	**0.118**	0.019
		Masked	$\alpha = 0.0$	**0.604**	0.248	**0.120**	0.059
	Undirected	Causal	$\alpha = 0.0$	**0.670**	0.109	**0.157**	0.026
			$\alpha = 0.1$	**0.679**	0.109	**0.171**	0.026
		Masked	$\alpha = 0.0$	**0.603**	0.260	**0.155**	0.080
			$\alpha = 0.1$	**0.647**	0.260	**0.173**	0.080
TRex	Directed	Causal	$\alpha = 0.0$	**0.650**	0.038	**0.131**	0.022
		Masked	$\alpha = 0.0$	**0.564**	0.248	**0.179**	0.078
	Undirected	Causal	$\alpha = 0.0$	**0.647**	0.040	**0.215**	0.027
			$\alpha = 0.1$	**0.704**	0.040	**0.302**	0.027
		Masked	$\alpha = 0.0$	**0.600**	0.248	**0.220**	0.142
			$\alpha = 0.1$	**0.666**	0.248	**0.272**	0.142
Twitter	Directed	Causal	$\alpha = 0.0$	**0.319**	0.035	**0.048**	0.019
		Masked	$\alpha = 0.0$	**0.270**	0.084	**0.049†**	0.047
	Undirected	Causal	$\alpha = 0.0$	**0.317**	0.036	**0.041**	0.018
			$\alpha = 0.1$	**0.301**	0.036	**0.048**	0.018
		Masked	$\alpha = 0.0$	**0.300**	0.083	0.037	**0.044†**
			$\alpha = 0.1$	**0.226**	0.083	**0.052**	0.044

Retrieval


Pubmed
Masked
Causal
0.0
0.2
0.4
0.6
0.8
2
4
6
8
10
k
Directed Baseline
Undirected Baseline
Directed, $\alpha = 0$
Undirected, $\alpha = 0$
Undirected, $\alpha = 0.1$



Trex
Masked
Causal
Directed Baseline
Undirected Baseline
Directed, $\alpha = 0$
Undirected, $\alpha = 0$
Undirected, $\alpha = 0.1$
k
1.0
0.8
0.6
0.4
0.2
0.0
2
4
6
8
10


Twitter
Masked
Causal
Directed Baseline
Undirected Baseline
Directed, $\alpha = 0$
Undirected, $\alpha = 0$
Undirected, $\alpha = 0.1$
0.12
0.10
0.08
0.06
0.04
0.02
0.00
2
4
6
8
10
k