Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Alexander Panchenko

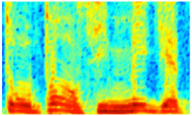
FROM UNSUPERVISED INDUCTION OF LINGUISTIC STRUCTURES FROM TEXT TOWARDS APPLICATIONS IN DEEP LEARNING

Graph embeddings

Text: sparse symbolic representation

AUDIO	IMAGES	TEXT

AUDIO



Audio Spectrogram

DENSE

IMAGES

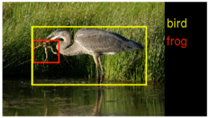


Image pixels

DENSE

TEXT

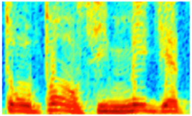
0	0	0	0.2	0	0.7	0	0	0	...	...

Word, context, or
document vectors

SPARSE

Text: sparse symbolic representation

AUDIO



Audio Spectrogram

DENSE

IMAGES

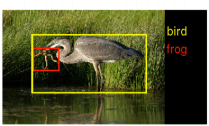
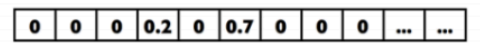


Image pixels

DENSE

TEXT



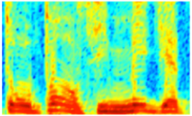
Word, context, or
document vectors

SPARSE

Image source:
`https://www.tensorflow.org/tutorials/word2vec`

Graph: sparse symbolic representation

AUDIO



Audio Spectrogram

DENSE

IMAGES

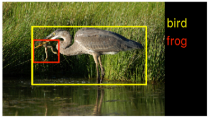
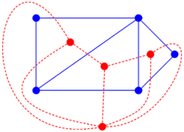


Image pixels

DENSE

GRAPH

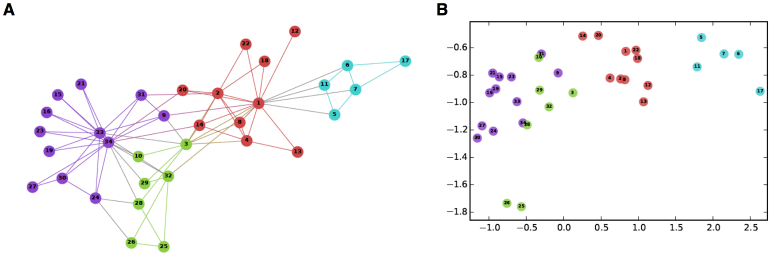


Nodes, edges, weights

SPARSE

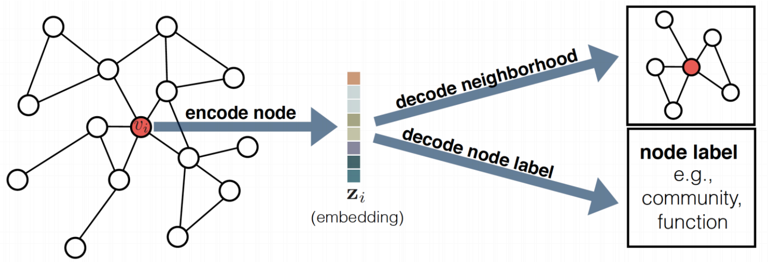
Embedding graph into a vector space

From a **survey on graph embeddings** [Hamilton et al., 2017]:



Learning with an “autoencoder”

From a **survey on graph embeddings** [Hamilton et al., 2017]:



Some established approaches

From a **survey on graph embeddings** [Hamilton et al., 2017]:

Type	Method	Decoder	Similarity measure	Loss function (ℓ)
Matrix factorization	Laplacian Eigenmaps [4]	$\|\mathbf{z}_i - \mathbf{z}_j\|_2^2$	general	$\mathrm{DEC}(\mathbf{z}_i, \mathbf{z}_j) \cdot s_{\mathcal{G}}(v_i, v_j)$
	Graph Factorization [1]	$\mathbf{z}_i^\top \mathbf{z}_j$	$\mathbf{A}_{i,j}$	$\|\mathrm{DEC}(\mathbf{z}_i, \mathbf{z}_j) - s_{\mathcal{G}}(v_i, v_j)\|_2^2$
	GraRep [9]	$\mathbf{z}_i^\top \mathbf{z}_j$	$\mathbf{A}_{i,j}, \mathbf{A}_{i,j}^2, ..., \mathbf{A}_{i,j}^k$	$\|\mathrm{DEC}(\mathbf{z}_i, \mathbf{z}_j) - s_{\mathcal{G}}(v_i, v_j)\|_2^2$
	HOPE [45]	$\mathbf{z}_i^\top \mathbf{z}_j$	general	$\|\mathrm{DEC}(\mathbf{z}_i, \mathbf{z}_j) - s_{\mathcal{G}}(v_i, v_j)\|_2^2$
Random walk	DeepWalk [47]	$\frac{e^{\mathbf{z}_i^\top \mathbf{z}_j}}{\sum_{k \in \mathcal{V}} e^{\mathbf{z}_i^\top \mathbf{z}_k}}$	$p_{\mathcal{G}}(v_j \mid v_i)$	$-s_{\mathcal{G}}(v_i, v_j) \log(\mathrm{DEC}(\mathbf{z}_i, \mathbf{z}_j))$
	node2vec [28]	$\frac{e^{\mathbf{z}_i^\top \mathbf{z}_j}}{\sum_{k \in \mathcal{V}} e^{\mathbf{z}_i^\top \mathbf{z}_k}}$	$p_{\mathcal{G}}(v_j \mid v_i)$ (biased)	$-s_{\mathcal{G}}(v_i, v_j) \log(\mathrm{DEC}(\mathbf{z}_i, \mathbf{z}_j))$

Graph embeddings using similarities

An submitted joint work with Andrei Kutuzov and Chris Biemann:

- Given a tree (V, E)
- **Leackock-Chodorow (LCH)** similarity measure:

$$sim(v_i, v_j) = -\log \frac{shortest\_path\_distance(v_i, v_j)}{2h}$$

Graph embeddings using similarities

An submitted joint work with Andrei Kutuzov and Chris Biemann:

- Given a tree (V, E)
- **Leackock-Chodorow (LCH)** similarity measure:

$$sim(v_i, v_j) = -\log \frac{shortest\_path\_distance(v_i, v_j)}{2h}$$

- **Jiang-Conrath (JCN)** similarity measure:

$$sim(v_i, v_j) = 2 \frac{\ln P_{lcs}(v_i, v_j)}{\ln P(v_i) + \ln P(v_j)}$$

Graph embeddings using similarities

path2vec: Approximating **Structural Node Similarities** with **Node Embeddings**:

$$J = \frac{1}{|T|} \sum_{(v_i, v_j) \in T} (\mathbf{v}_i \cdot \mathbf{v}_j - sim(v_i, v_j))^2,$$

where:

- $sim(v_i, v_j)$ - the value of a 'gold' similarity measure between a pair of nodes v_i and v_j;
- $\mathbf{v}_i$ - an embeddings of node;
- T - training batch.

Speedup: graph vs embeddings

Computation of 82,115 pairwise similarities:

Model	*Running time*
LCH in NLTK	30 sec.
JCN in NLTK	6.7 sec.
FSE embeddings	0.713 sec.
path2vec and other float vectors	**0.007** sec.

Results: goodness of fit

Spearman correlation scores with WordNet similarities on SimLex999 noun pairs:

	Selection of synsets		
Model	JCN-SemCor	JCN-Brown	LCH
WordNet	1.0	1.0	1.0
Node2vec	0.655	0.671	0.724
Deepwalk	0.775	0.774	0.868
FSE	0.830	0.820	0.900
path2vec	**0.917**	**0.914**	**0.934**

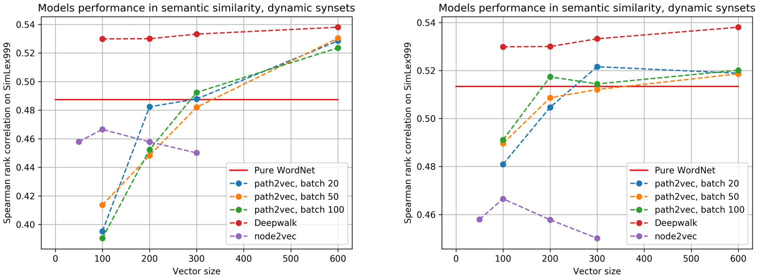
Results: SimLex999 dataset

Spearman correlations with human SimLex999 noun similarities:

Model	*Correlation*
Raw WordNet JCN-SemCor	0.487
Raw WordNet JCN-Brown	0.495
Raw WordNet LCH	0.513
node2vec [Grover & Leskovec, 2016]	0.450
Deepwalk [Perozzi et al., 2014]	0.533
FSE [Subercaze et al., 2015]	**0.556**
path2vec JCN-SemCor	0.526
path2vec JCN-Brown	0.487
path2vec LCH	0.522

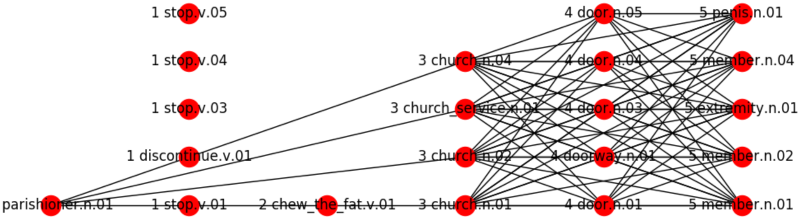
Results: SimLex999 dataset

JCN (left) and LCH (right):



Results: word sense disambiguation

WSD: each column lists all the possible synsets for the corresponding word.



Results: word sense disambiguation

F1 scores on Senseval-2 word sense disambiguation task:

Model	F-measure		
WordNet JCN-SemCor	**0.620**		
WordNet JCN-Brown	0.561		
WordNet LCH	0.547		
node2vec [Grover & Leskovec, 2016]	0.501		
Deepwalk [Perozzi et al., 2014]	0.528		
FSE [Subercaze et al., 2015]	0.536		
path2vec			
Batch size:	20	50	100
JCN-SemCor	**0.543**	**0.543**	0.535
JCN-Brown	0.538	0.515	0.542
LCH	0.540	0.535	0.536

Grover, A. & Leskovec, J. (2016).
Node2vec: Scalable feature learning for networks.
In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855–864).: ACM.

Hamilton, W. L., Ying, R., & Leskovec, J. (2017).
Representation learning on graphs: Methods and applications.
IEEE Data Engineering Bulletin, September 2017.

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014).
Deepwalk: Online learning of social representations.
In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701–710).: ACM.

Subercaze, J., Gravier, C., & Laforest, F. (2015).
On metric embedding for boosting semantic similarity computations.

In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 8–14).: Association for Computational Linguistics.