The representation of text in a multilayer complex network for deep learning

Karlo Babić, Sanda Martinčić-Ipšić
Department of Informatics, University of Rijeka
{karlo.babic, smarti}@inf.uniri.hr

LANGUAGE NETWORKS

Introduction

Language can be modeled via **complex networks**

- each word is a node and interactions amongst words are links
- allows systematic quantitive analyses

Model the various **language levels**

Deepening the understanding of conceptual similarities, differences and universalities in natural languages

Establish a bridge:

- linguistics, complex network science, computer science, and natural language processing

Multilayer Network

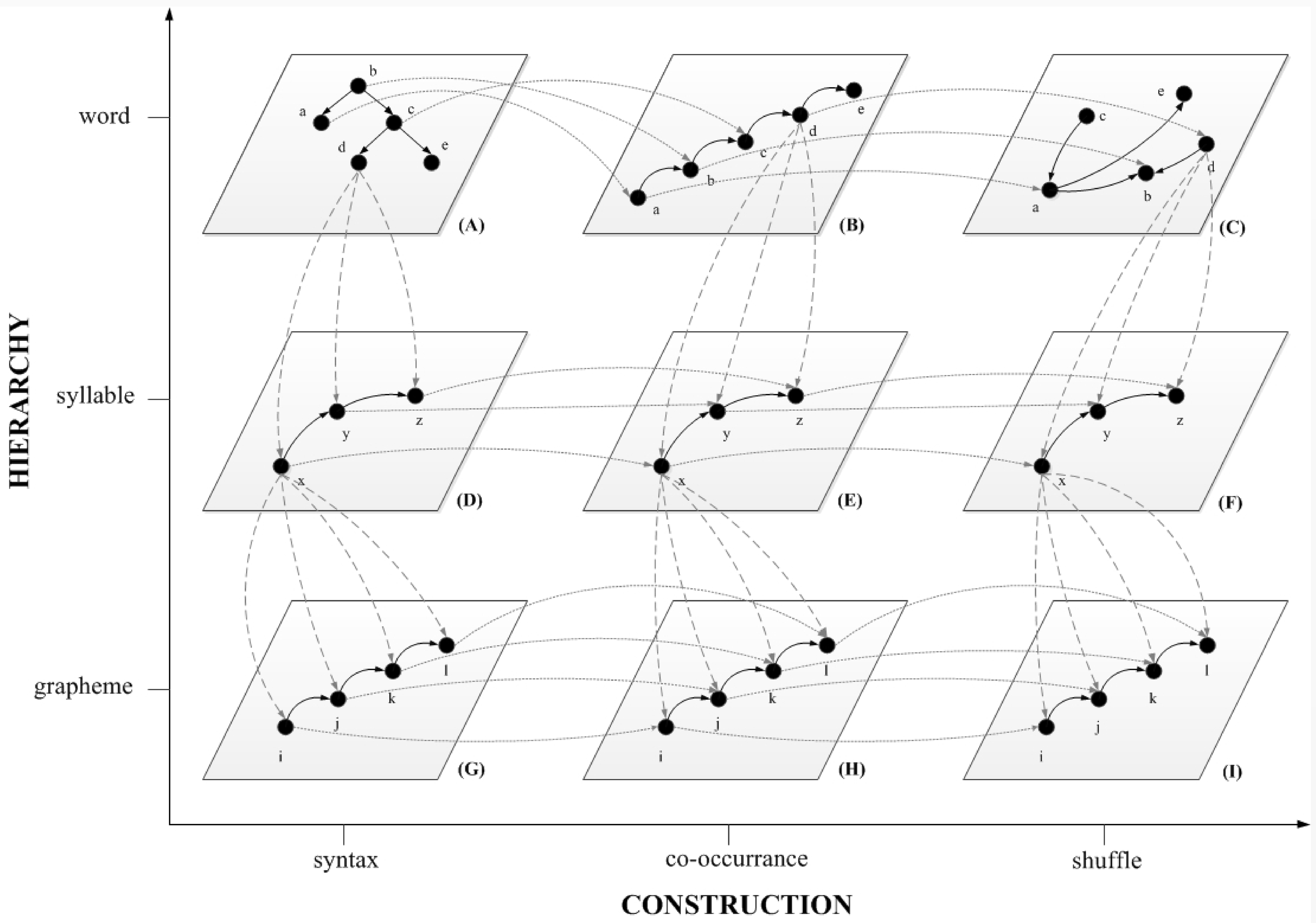
Various **levels:**

paragraph level:
- co-occurance

sentence level:
- co-occurance

word level:
- co-occurance
- syntax
- semantics

sub-word level:
- morphology (morphosyntactic)
- syllabic
- phonetic (phonology)
- graphemic



Language networks can be viewed through different **perspectives:**

- different levels (e.g. sentence-level, word-level, subword-level),
- different construction rules (e.g. co-occurrence, shuffle),
- different languages (e.g. English, Spanish, Croatian)

Multilayer Network Definition

Multilayer Language Network M is a quintuple $\mathbf{M = (V_M, E_M, V, L, C)}$

- V is a non-empty set of nodes,
- C is a non-empty set of perspective elements,
- L is a set of perspects L_i where $\{L_0, L_1, L_2\}$ is a partition of C.
    - L_0 - language perspect, L_1 - hierarchy perspect, and L_2 - construction perspect.
- For perspect $L_1 = \{g_1, \dots, g_k\}$ sequence of its elements $g_1, \dots, g_k$ is the subsequence of the following sequence - hierarchy:
    - discourse, sentence, phrase, syntagm, word, morphem, syllable, phoneme, grapheme

An element of the set $L_0 \times L_1 \times L_2$ is called a **layer**,

$V_M \subseteq V \times L_0 \times L_1 \times L_2$ is the set whose elements are called **MLN-nodes**,

$E_M \subseteq V_M \times V_M$ is the set of **edges**.

This work has been supported by the University of Rijeka under the project number **uniri-drustv-18-20** and by **COST Action CA15109 COSTNET**.

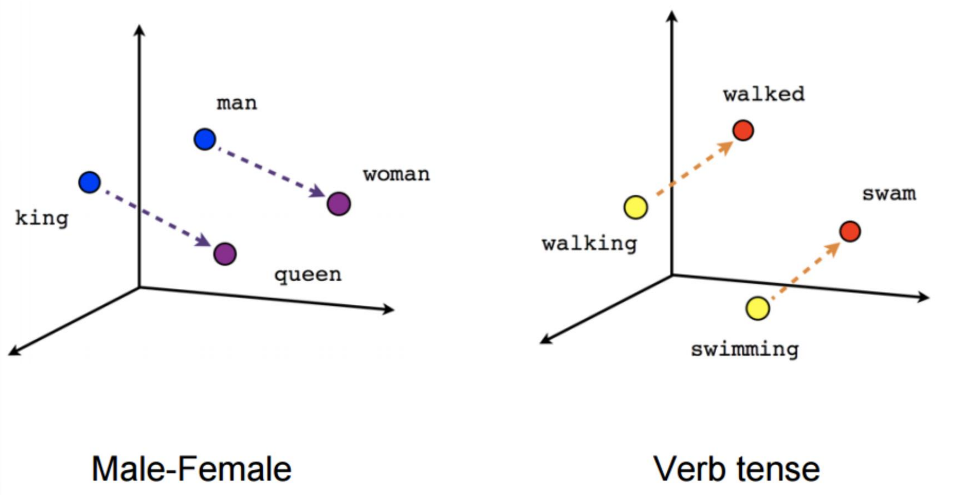
EMBEDDING LANGUAGE NETWORKS

Representation of text

Text embedding: word2vec (deep learning)

Keeps meaningful relationships

Embedding can be used for semantic analysis, text summarization, or similar applications



Language networks as input

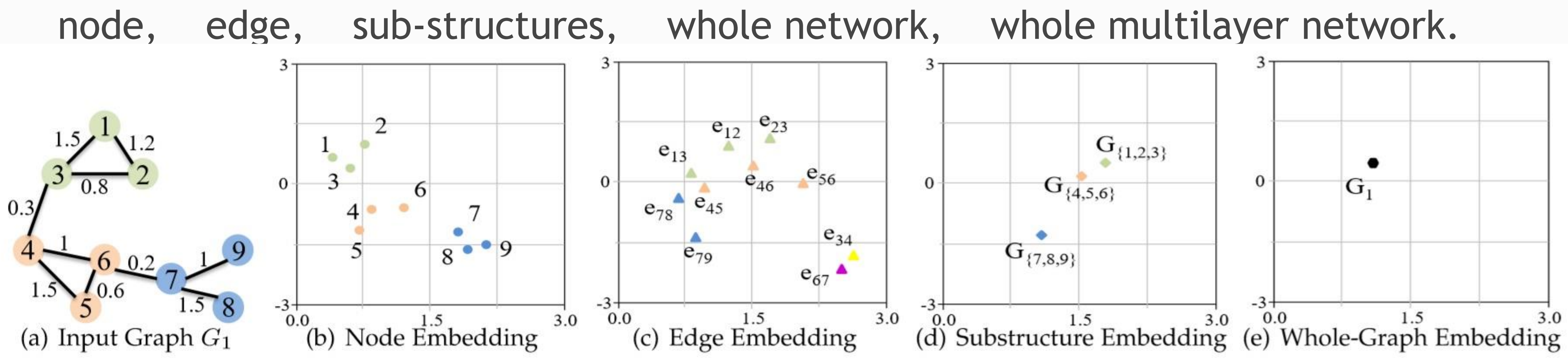
People process language on **multiple levels of abstraction**.

Letters -> words -> sentences -> text

Embedding of text: keeps all levels

Types of network embedding methods:

node, edge, sub-structures, whole network, whole multilayer network.



Node embedding techniques

Matrix factorization:

- embedding is created by factorizing the matrix which contains pairwise similarities
- pairwise node similarities are preserved in the embedding

Deep learning (Deepwalk):

- treat nodes as words, generate short random walks,
- skip-gram can be applied on these walks to obtain embeddings

Edge reconstruction:

- creates embedding by optimizing edge reconstruction
- edges are generated based on node embedding

Network embedding techniques

Sub-structures:

- sum all node embeddings
- calculate embedding of a dummy node

Graph kernels:

- embedding is a vector containing counts of elementary substructures
- substructures can be detected with kernels:
  graphlets, subtree patterns, random walks

Dissimilarity space embedding:

- chooses n prototypes (graphs used as base for embeddings)
- a graph is than embedded by calculating dissimilarity measure for every prototype graph
- embedding is a vector, every element is a distance from one prototype

Deep learning (graph2vec):

- graph2vec neural network creates embeddings for graphs by learning to predict subgraphs for every graph in training set

Conclusion

Multilayer network representation could be very useful for usage as input for **machine learning**

It could be used for:

text type recognition, text similarity, information retrieval, other NLP tasks

Sources

Kirigin, Tajana Ban, Ana Meštrović, and Sanda Martinčić-Ipšić. "Towards a formal model of language networks." *International Conference on Information and Software Technologies*. Springer, Cham, 2015.

Yang, Cheng, et al. "Network representation learning with rich text information." *IJCAI*. 2015.

Perozzi, Bryan, Rami Al-Rfou, and Steven Skiena. "Deepwalk: Online learning of social representations." *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014.

Bordes, Antoine, et al. "Translating embeddings for modeling multi-relational data." *Advances in neural information processing systems*. 2013

Yanardag, Pinar, and S. V. N. Vishwanathan. "Deep graph kernels." *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015.

Bunke, Horst, and Kaspar Riesen. "Graph classification based on dissimilarity space embedding." *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, Berlin, Heidelberg, 2008.

Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, YangLiu, and Shantanu Jaiswal. graph2vec: Learning distributed representations of graphs.arXivpreprintarXiv:1707.05005, 2017.