CANE: Context-Aware Network Embedding for Relation Modeling

MACL 2017

Cunchao Tu
Tsinghua University

Han Liu
Northeastern University

Zhiyuan Liu
Tsinghua University

Maosong Sun
Tsinghua University

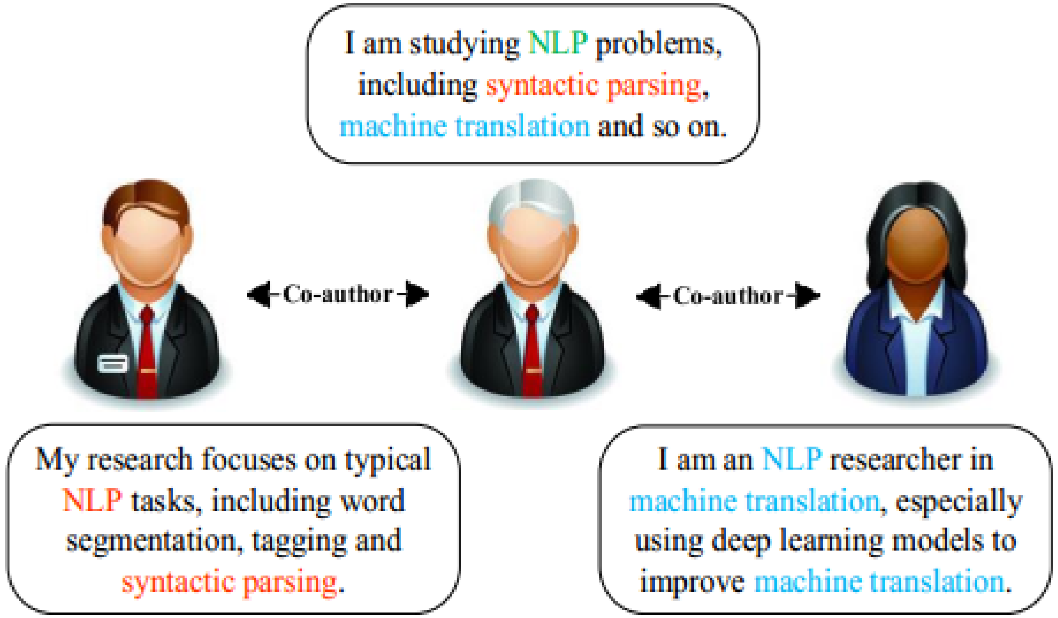
● 引言

most existing NE methods only arrange one single embedding vector to each vertex, and give rise to the following two invertible issues:

(1)These methods cannot flexibly cope with the aspect transition of a vertex when interacting with different neighbors.

(2)In these models, a vertex tends to force the embeddings of its neighbors close to each other, which may be not the case all the time.

To address these issues, we aim to propose a **C**ontext-**A**ware **N**etwork **E**mbedding (CANE) framework for modeling relationships between vertices precisely. More specifically, we present CANE on information networks, where each vertex also contains rich external information such as text, labels or other meta-data, and the significance of context is more critical for NE in this scenario.



● 引言

● context-free embedding

It means the embedding of a vertex is fixed and won’t change with respect to its context information

● context-aware embedding

CANE learns various embeddings for a vertex according to its different contexts.

the **selective attention scheme** and build **mutual attention** between u and v into models

● 相关工作

We first give basic notations and definitions in this work. Suppose there is an information network $G = (V, E, T)$, where V is the set of vertices, $E \subseteq V \times V$ are edges between vertices, and T denotes the text information of vertices. Each edge $e_{u,v} \in E$ represents the relationship between two vertices (u, v), with an associated weight $w_{u,v}$. Here, the text information of a specific vertex $v \in V$ is represented as a word sequence $S_v = (w_1, w_2, \ldots, w_{n_v})$, where $n_v = |S_v|$. NRL aims to learn a low-dimensional embedding $\mathbf{v} \in \mathbb{R}^d$ for each vertex $v \in V$ according to its network structure and associated information, e.g. text and labels. Note that, $d \ll |V|$ is the dimension of representation space.

- 提出的框架

we propose two types of embeddings for a vertex *v*, i.e., structurebased embedding $\mathbf{v}^s$ and text-based embedding $\mathbf{v}^t$.

we can simply **concatenate** them and obtain the vertex embeddings as $\mathbf{v} = \mathbf{v}s \oplus \mathbf{v}t$,

● 提出的框架

With above definitions, CANE aims to maximize the overall objective of edges as follows:

$$\mathcal{L} = \sum_{e \in E} L(e). \tag{1}$$

$$L(e) = L_s(e) + L_t(e), \tag{2}$$

where $L_s(e)$ denotes the structure-based objective and $L_t(e)$ represents the text-based objective.

- 提出的框架

$$L(e) = \boxed{L_s(e)} + L_t(e), \tag{2}$$

Thus, the structure-based objective aims to measure the log-likelihood of a directed edge using the structure-based embeddings as

$$L_s(e) = w_{u,v} \log p(\mathbf{v}^s|\mathbf{u}^s). \tag{3}$$

Following LINE (Tang et al., 2015), we define the conditional probability of v generated by u in Eq. (3) as

$$p(\mathbf{v}^s|\mathbf{u}^s) = \frac{\exp(\mathbf{u}^s \cdot \mathbf{v}^s)}{\sum_{z \in V} \exp(\mathbf{u}^s \cdot \mathbf{z}^s)}. \tag{4}$$

- 提出的框架

$$L(e) = L_s(e) + \boxed{L_t(e)}, \qquad (2)$$

The text-based objective $L_t(e)$ can be defined with various measurements. To be compatible with $L_s(e)$, we define $L_t(e)$ as follows:

$$L_t(e) = \alpha \cdot L_{tt}(e) + \beta \cdot L_{ts}(e) + \gamma \cdot L_{st}(e), \quad (5)$$

where α, β and γ control the weights of various parts, and

$$\begin{aligned} L_{tt}(e) &= w_{u,v} \log p(\mathbf{v}^t|\mathbf{u}^t), \\ L_{ts}(e) &= w_{u,v} \log p(\mathbf{v}^t|\mathbf{u}^s), \\ L_{st}(e) &= w_{u,v} \log p(\mathbf{v}^s|\mathbf{u}^t). \end{aligned} \qquad (6)$$

● 文本嵌入的方法（CNN）

Looking-up. Given a word sequence $S = (w_1, w_2, \ldots, w_n)$, the looking-up layer transforms each word $w_i \in S$ into its corresponding word embedding $\mathbf{w}_i \in \mathbb{R}^{d'}$ and obtains embedding sequence as $\mathbf{S} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n)$. Here, d' indicates the dimension of word embeddings.

Convolution. After looking-up, the convolution layer extracts local features of input embedding sequence $\mathbf{S}$. To be specific, it performs convolution operation over a sliding window of length l using a convolution matrix $\mathbf{C} \in \mathbb{R}^{d\times(l\times d')}$ as follows:

$$\mathbf{x}_i = \mathbf{C} \cdot \mathbf{S}_{i:i+l-1} + \mathbf{b}, \tag{7}$$

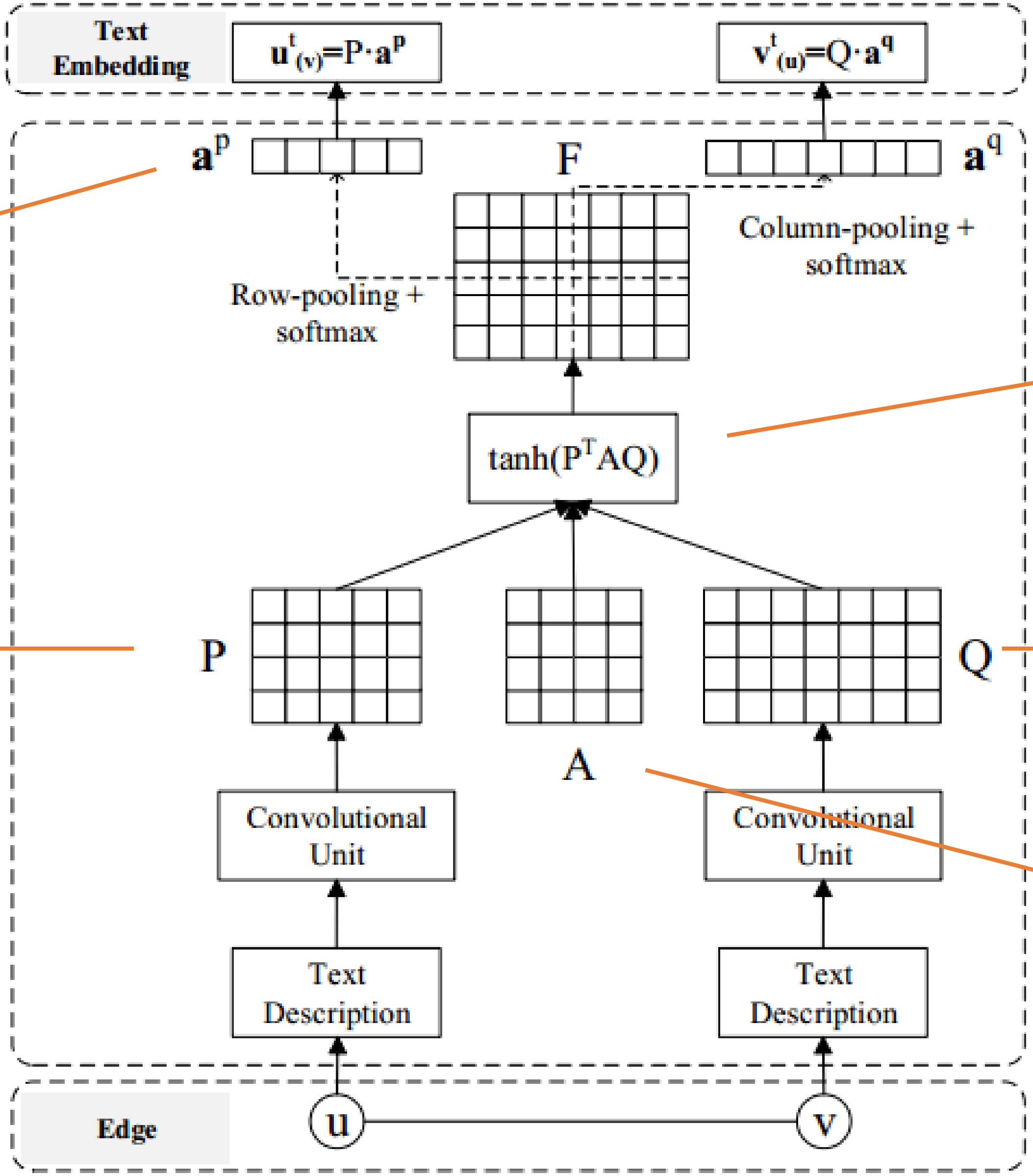
- 文本嵌入的方法（CNN）

Max-pooling. To obtain the text embedding $\mathbf{v}^t$, we operate max-pooling and non-linear transformation over $\{\mathbf{x}_0^i, \ldots, \mathbf{x}_n^i\}$ as follows:

$$r_i = \tanh(\max(\mathbf{x}_0^i, \ldots, \mathbf{x}_n^i)), \quad (8)$$

At last, we encode the text information of a vertex with CNN and obtain its text-based embedding $\mathbf{v}^t = [r_1, \ldots, r_d]^T$. As $\mathbf{v}^t$ is irrelevant to the other vertices it interacts with, we name it as context-free text embedding.

● 文本嵌入的方法（CNN）



$$a_i^p = \frac{\exp(g_i^p)}{\sum_{j\in[1,m]}\exp(g_j^p)}.$$

$$\mathbf{u}_{(v)} = \mathbf{u}^s \oplus \mathbf{u}^t_{(v)} \text{ and } \mathbf{v}_{(u)} = \mathbf{v}^s \oplus \mathbf{v}^t_{(u)}$$

correlation matrix F

$\mathbf{P} \in \mathbb{R}^{d\times \boxed{m}}$

$\mathbf{Q} \in \mathbb{R}^{d\times \boxed{n}}$

the lengths of two corresponding text sequences

Attentive matrix

- CANE的优化

According to Eq. (3) and Eq. (6), CANE aims to maximize several conditional probabilities between $\mathbf{u} \in \{\mathbf{u}^s, \mathbf{u}^t_{(v)}\}$ and $\mathbf{v} \in \{\mathbf{v}^s, \mathbf{v}^t_{(u)}\}$. It is intuitive that optimizing the conditional probability using softmax function is computationally expensive. Thus, we employ negative sampling (Mikolov et al., 2013b) and transform the objective into the following form:

$$\log \sigma(\mathbf{u}^T \cdot \mathbf{v}) + \sum_{i=1}^{k} E_{z \sim P(v)} [\log \sigma(-\mathbf{u}^T \cdot \mathbf{z})], \quad (13)$$

● 实验

Datasets

Datasets	**Cora**	**HepTh**	**Zhihu**
#Vertices	2,277	1,038	10,000
#Edges	5,214	1,990	43,894
#Labels	7	–	–

Baselines

Structure-only
Structure and Text

- 应用场景
  - 链路预测
  - 顶点分类
  - 案例学习
- 未来工作
  - 用于其他信息网络
  - 整合和预测NE中顶点之间的关系

Edge #1: (A, B)

Machine Learning research making great progress many directions This article summarizes four directions discusses current open problems The four directions improving classification accuracy learning ensembles classifiers methods scaling supervised learning algorithms reinforcement learning learning complex stochastic models

The problem making optimal decisions uncertain conditions central Artificial Intelligence If state world known times world modeled Markov Decision Process MDP MDPs studied extensively many methods known determining optimal courses action policies The realistic case state information partially observable Partially Observable Markov Decision Processes POMDPs received much less attention The best exact algorithms problems inefficient space time We introduce Smooth Partially Observable Value Approximation SPOVA new approximation method quickly yield good approximations improve time This method combined reinforcement learning methods combination effective test cases

Edge #2: (A, C)

Machine Learning research making great progress many directions This article summarizes four directions discusses current open problems The four directions improving classification accuracy learning ensembles classifiers methods scaling supervised learning algorithms reinforcement learning learning complex stochastic models

In context machine learning examples paper deals problem estimating quality attributes without dependencies among Kira Rendell developed algorithm called RELIEF shown efficient estimating attributes Original RELIEF deal discrete continuous attributes limited twoclass problems In paper RELIEF analysed extended deal noisy incomplete multiclass data sets The extensions verified various artificial one well known realworld problem

谢 谢 大 家