

# K-BERT(AAAI2020)

2020年11月18日 16:53

## Introduction

Two Challenges:

(1) Heterogeneous Embedding Space (HES)

通常情况下，词向量的embedding和知识图谱里的实体之间的向量空间是相互独立的，这导致如果直接混合知识图谱和词向量，两者是不匹配的。

(2) Knowledge Noise (KN)

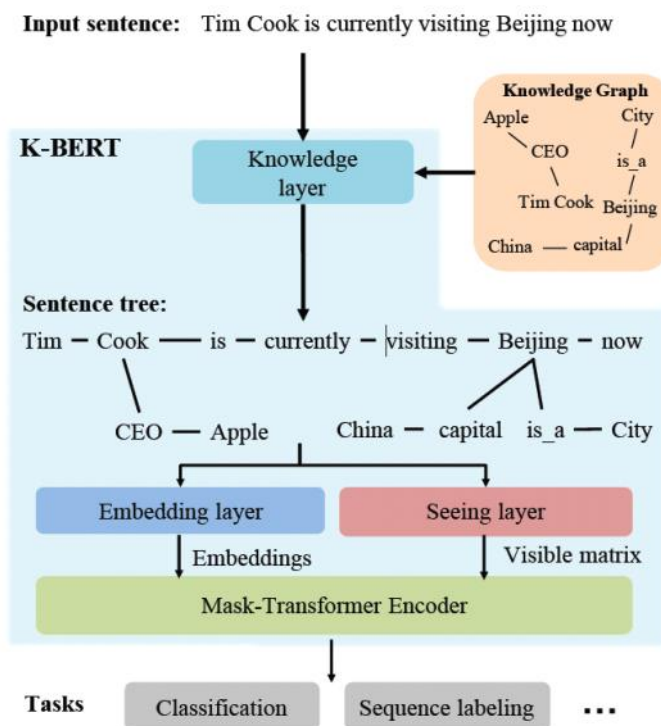
如果预训练过程中混入大量的知识，会扰乱训练语料的正确语义。

## Main contributions

- (1) Proposed a knowledge-enabled Language Representation model
- (2) Delicate Injection of KG. Model outperforms BERT in open-domain and domain-specific tasks
- (3) Create a knowledge graphs publicly available

## 相关工作

- (1) ERNIE: 融合实体信息做MLM预训练任务，但是忽略了实体之间的关系。
- (2) COMET: 将知识图谱的三元组形式作为语料对GPT进行常识学习，但是训练过程是非常低效的。



## 一些定义

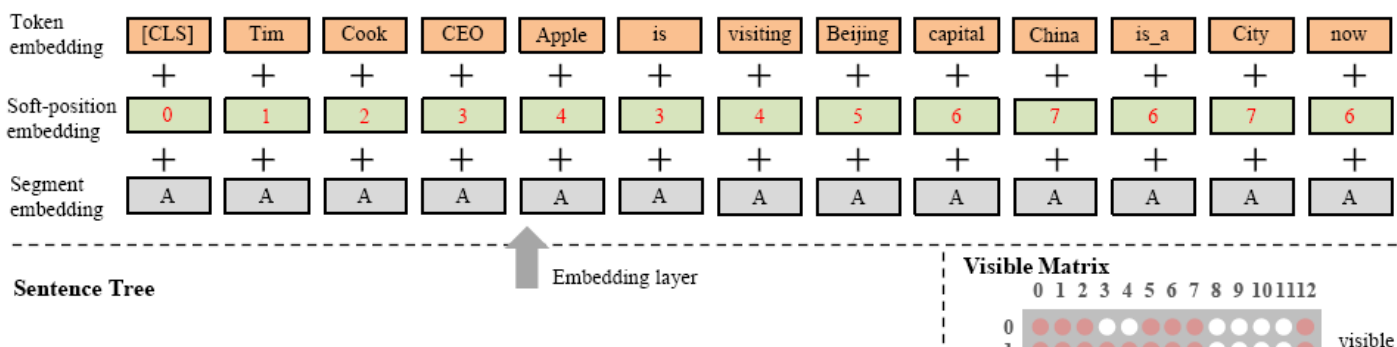
A sequence of tokens:  $s = \{w_0, w_1, w_2, \dots, w_n\}$

知识图谱的三元组形式  $\varepsilon = (w_i, r_j, w_k)$

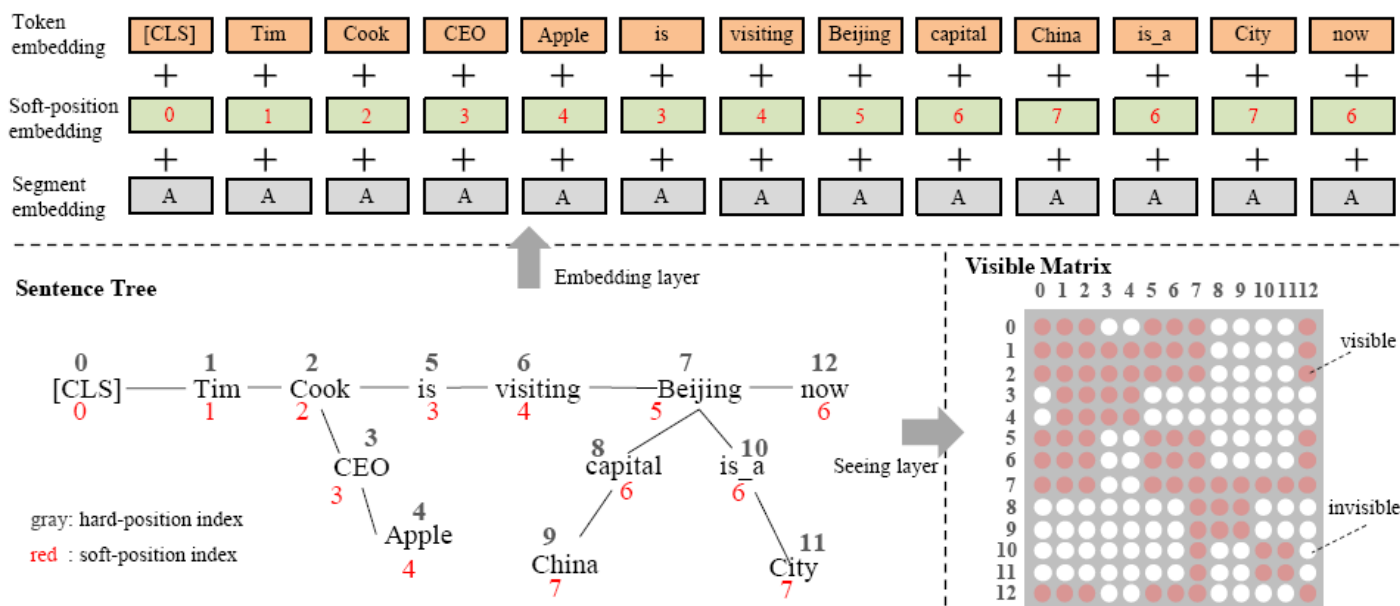
## 模型结构:

- knowledge layer
- Embedding layer
- seeing layer
- mask-transformer

## Embedding Representation



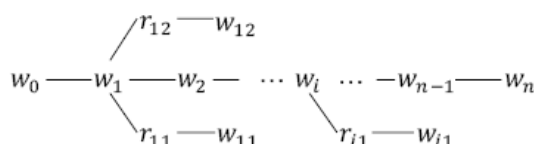
## Embedding Representation



Knowledge layer

分为两步

- (1) 查询和句子单词相关的三元组
- (2) 利用三元组建立句子树



Embedding layer

目标:将句子树转化为embedding representation之后以便能够输入到transformer中

- (1) Token embedding

这里使用的是BERT相同的Vocabulary,使用相同的[CLS]和[MASK]标记,使用的token输入方法是把分支的token输入到父节点之后,后续的句子顺延。

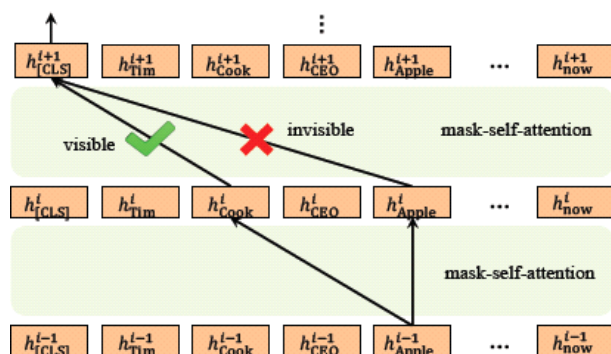
- (2) Soft-position embedding

如果单纯使用硬编码,会导致一个问题,比如[is]的编码是5, [COOK]的编码是2, [APPLE]编码是4, 如果直接将知识插入的话, 模型可能将[is]的主语看做[Apple], 因此引入软位置编码, 任何一个分支的编码从父节点重新计数, 由于软位置编码带来的编码冲突 ([is],[CEO]都是3) 的问题可以由后面的注意力MASK矩阵解决。

- (3) Segment embedding

和BERT相同, 同一个句子的segment embedding相同

Seeing layer



$$M_{ij} = \begin{cases} 0 & w_i \ominus w_j \\ -\infty & w_i \oslash w_j \end{cases}$$

其中i和j为硬位置编码

#### Mask-Self-Attention

$$Q^{i+1}, K^{i+1}, V^{i+1} = h^i W_q, h^i W_k, h^i W_v$$

$$S^{i+1} = \text{softmax}\left(\frac{Q^{i+1} K^{i+1 \top} + M}{\sqrt{d_k}}\right)$$

$$h^{i+1} = S^{i+1} V^{i+1}$$

例如[cls]虽然不能直接获得[apple]的知识，但是[Cook]的嵌入表示中隐藏了[apple]的信息。

#### Experiments