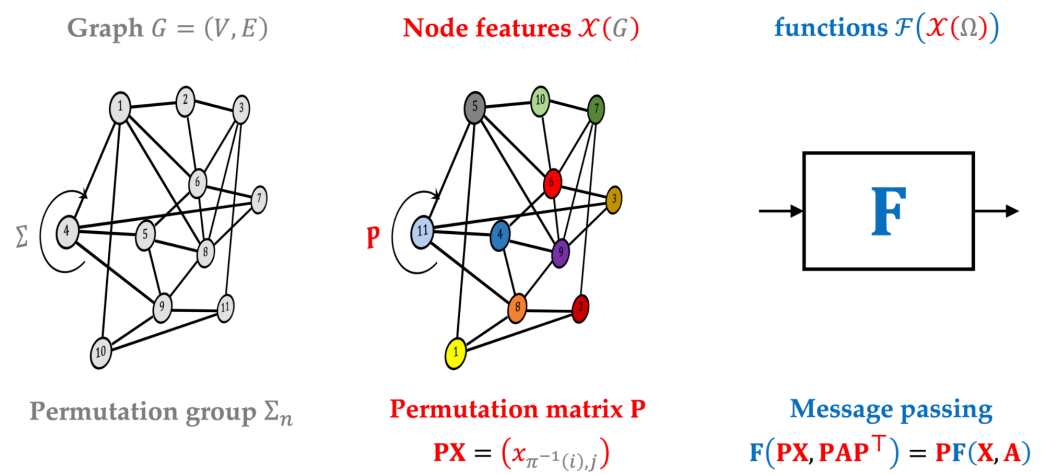
语言模型不变量

杨天骥　2024/5/14

几何深度学习

例子：输入 X（节点特征），A（邻接矩阵）[1]



[1] https://blog.x.com/engineering/en_us/topics/insights/2021/graph-neural-networks-through-the-lens-of-differential-geometry

语言模型的平移不变量

本文主要研究语言的平移不变量，它要求用于估计一个序列（句子）在语言当中出现概率的模型 π 具备如下性质：

$$\sum_{x_1 \in \Sigma} \pi(x_1, x_2, ..., x_n) = \pi(x_2, ..., x_n)$$

其中 Σ 是词表的大小，而 $x_2, ..., x_n$ 是一句句子。

引入平移不变量即假设我们不知道这个训练数据在整个语言中扮演的语素。

训练：下一词元 -> 上一词元

在目前流行的预训练任务（下一词元预测）和模型架构下，要满足平移不变量是非常困难的。相关的结果在 3.2 节当中有所呈现，例如这个约束会导致线性 RNN 出现参数冻结。

根据定理 3.2 提供的启发，将训练任务设定为上一词元的概率估计。

$$f(x_1 \mid x_2, ..., x_n)$$

推理

实际使用模型进行推理时，仍然使用下一词元预测。本文提出一种通过贝叶斯公式计算下一词元概率的方法

$$p(x_{n+1} \mid x_1, ..., x_n) = \prod_j \frac{f(x_j \mid x_{j+1}, ..., x_n, x_{n+1})}{f(x_j \mid x_{j+1}, ..., x_n)}$$

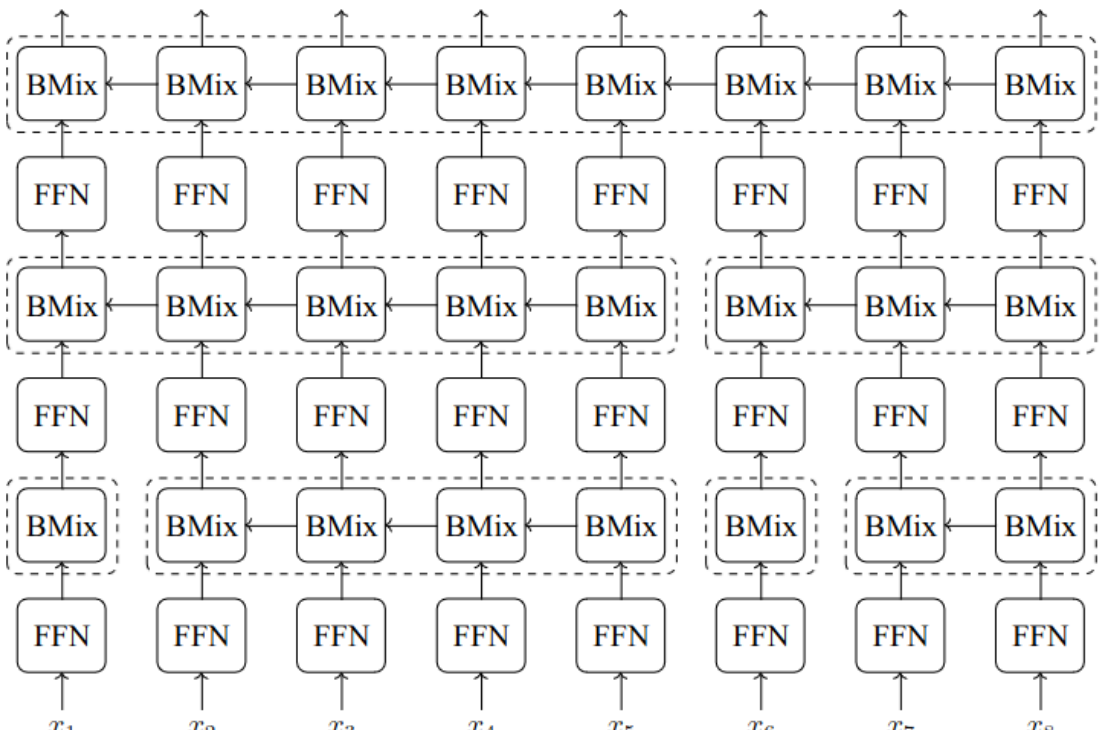
优化

根据已经提供的结果，模型能训练也能推理，但是存在效率问题。贝叶斯公式的计算需要模型在所有的形如 $x_{j+1}, ..., x_n, x_{n+1}$ 的句子上全部应用一遍，这样的计算代价就太大了（$O\left(\sum_{m<n} F(m)\right)$，其中 $F(m)$ 是在长度为 m 的句子上的计算代价）。

本文根据如下假设提供了优化：

- 对多数下标 j, $f\left(x_j \mid x_{j+1}, ..., x_n, x_{n+1}\right) = f\left(x_j \mid x_{j+1}, ..., x_n\right)$

优化



实验

普通数据集(WikiText, Shakespeare): 各自选择了 4 万个长度为 1024 字符的序列进行训练。评估时使用同样长度的序列。

数据增强(WikiText(Ex), Shakespeare(Ex))：从中各自抽取 4 万对长度均为 128 字符且有 32 个字符相交的序列进行训练。随后，从剩余数据中采取长为 128*2-32 的序列进行评估。

实验

实验的结果如下：

Model/Task	WikiText（Ex）	WikiText	Shakespeare（Ex）	Shakespeare
BMix	**4.38**	2.85	**4.56**	**2.79**
Attention	5.43	**2.83**	5.45	2.91