语言模型概述

概要

- 统计语言模型
- 神经网络语言模型
- 预训练语言模型

语言模型历史

- 统计语言模型 (~2010s)
  - 核心思想是基于计数和概率统计，直接对词序列的表面形式进行建模
    - N-gram 模型
- 神经网络语言模型 (2010s ~ 2017)
  - 引入了分布式表示，即用连续的向量来表示词和上下文
    - 词嵌入 (Word Embeddings) - 语言模型的“副产品”与基石
    - 基于前馈神经网络的语言模型 (FFNN-LM)
    - 基于循环神经网络的语言模型 (RNN-LM)
    - 序列到序列模型 (Seq2Seq) - 语言模型的“应用与扩展”
- 预训练语言模型(2018 ~至今)
  - 预训练 + 微调：模型先在海量无标签数据上进行预训练，然后再在特定的下游任务上进行微调
    - 基于Transformer的语言模型
    - 大规模预训练模型 (Large-scale Pre-trained Models)
      - 自编码模型 (Auto-Encoding Models)，代表: BERT
      - 自回归模型 (Auto-Regressive Models)，代表: GPT (Generative Pre-trained Transformer) 系列
      - 编码器-解码器模型 (Encoder-Decoder Models)，代表: T5, BART

什么是语言模型

什么是语言模型（ Language Model, LM） ？

- 语言模型是一个为自然语言序列（句子、段落等）分配概率的函数
  - 给定一个由 m 个词组成的序列 $W = (w_1, w_2, \dots, w_m)$，语言模型的任务就是计算这个序列作为一个整体出现的概率：$P(W) = P(w_1, w_2, \dots, w_m)$
- 概率反映了这个序列在特定语言中有多 " 自然 " 或多 " 合理 "
  - 好的语言模型会给符合语法、语义流畅的句子赋予高概率
    - P（ " 今天天气真不错 " ）$\gg P$（ " 不错天气今天真 " ）
  - 预测在给定前文的情况下，下一个最可能出现的词是什么。根据链式法则，概率可以分解为
    - $P(W) = P(w_1) \cdot P(w_2 \mid w_1) \cdot P(w_3 \mid w_1, w_2) \cdot \dots$
- 语言模型的核心任务也可以看作是计算下一个词的条件概率
  - $P(w_t \mid w_1, \dots, w_{t-1})$ 。

语言模型“能做什么”

- 从功能和任务的角度看，语言模型是一个能够理解和生成人类语言的计算模型
  - 理解
    - 评估文本质量：判断一个句子是否通顺、合乎逻辑（通过计算其概率）
    - 表征文本语义：将单词、句子或段落转换成能够捕捉其深层含义的数字向量（即嵌入或隐藏状态）。这是BERT等模型的核心能力
  - 生成
    - 预测下一个词：这是最基本的功能，也是所有生成任务的基础
    - 续写文本：给定一段开头，继续写下去，形成连贯的段落或文章
    - 条件生成：根据给定的指令、问题或输入文本，生成相应的输出文本（如翻译、回答问题、写摘要）
      - 现代聊天机器人（如ChatGPT）的核心能力

- 该定义解释了为什么语言模型是自然语言处理（NLP）领域几乎所有任务的基石

基础模型 (Foundation Model)

- 语言模型是海量文本数据上预训练，获得关于世界的大量通用知识和推理能力，并能通过自然语言接口与人类交互以完成广泛任务的基础模型
  - 基础模型: 通用基础，可通过微调、提示或上下文学习来适应不同下游任务
  - 涌现能力 (Emergent Abilities): 当模型规模（参数量、数据量）达到一定程度后，会“涌现”在小模型上不具备的复杂能力，如进行数学推理、编写代码、遵循复杂指令等
  - 世界知识: 不仅仅学习语言的语法规则，还从数据中隐式地学习大量关于现实世界的事实、概念和关系
  - 自然语言接口: 它允许非专业用户通过对话的方式来驱动复杂的计算和任务执行
- 该定义解释了为什么像GPT-4这样的现代语言模型能够产生如此巨大的影响，远远超出了传统NLP的范畴

评估标准：困惑度 (Perplexity)

- 一个好的语言模型，应该对真实的、自然的测试文本给予高概率
- 在测试集 $W = (w_1, \dots, w_N)$ 上，困惑度 (Perplexity, PPL) 定义为测试集逆几何平均概率

$$\text{PPL}(W) = P(w_1, \dots, w_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{P(w_1, \dots, w_N)}}$$

  - 其中 N 是测试集的总词数
- 等价形式 (使用交叉熵)

$$\text{PPL}(W) = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log P(w_i|w_1, \dots, w_{i-1})\right) = \exp(\text{Cross-Entropy})$$

- 解读
  - 困惑度可以被直观地理解为“模型在预测下一个词时，平均有多少个等可能性的选择”
    - 越少约好
    - 如果PPL=100，意味着模型在每个位置上，其不确定性等价于从100个词中均匀随机选择一个

统计语言模型

N-gram 模型

- 核心思想: 马尔可夫假设。一个词的出现只依赖于它前面 n-1 个词
- 方法: 通过在大型语料库中统计词组（n-grams）的频率来计算条件概率

- N-gram是第一个实用且被广泛应用的语言模型范式，统治了NLP领域数十年。它是后续所有模型要超越的基线
- 存在数据稀疏性、无法处理长期依赖、无语义泛化能力

神经网络语言模型

神经网络语言模型

- 词嵌入 (Word Embeddings) - 语言模型的“副产品”与基石
- 基于前馈神经网络的语言模型 (FFNN-LM)
- 基于循环神经网络的语言模型 (RNN-LM)
- 序列到序列模型 (Seq2Seq) - 语言模型的“应用与扩展”

词嵌入 (Word Embeddings)

- 词嵌入 (Word Embeddings) - 语言模型的“副产品”与基石
  - 核心思想: 一个词的意义由其上下文决定。将词映射到低维、连续的向量空间，使得语义相似的词在空间中距离相近
  - 代表: Word2Vec，GloVe, FastText

  - Word2Vec本身不是一个完整的语言模型（它的目标是学习词向量，而不是计算句子概率），但它是所有现代神经网络语言模型的基石。它解决了N-gram的语义泛化问题，是开启深度学习NLP时代的钥匙

基于前馈神经网络的语言模型

- 基于前馈神经网络的语言模型 (FFNN-LM)
  - 核心思想: 使用一个固定窗口大小（类似N-gram的n）的前文词向量，拼接后输入到一个前馈神经网络（FFNN）中，来预测下一个词
  - 代表: Bengio (2003) 的开创性工作
  - 首次将词嵌入和神经网络结合用于语言建模，是NNLM的鼻祖。但它仍受限于固定窗口，无法捕捉长期依赖

基于循环神经网络的语言模型 (RNN-LM)

- 基于循环神经网络的语言模型 (RNN-LM)
  - 核心思想: 引入循环结构，用一个动态更新的隐藏状态（Hidden State）来压缩和传递任意长度的历史信息。
  - 代表: RNN, LSTM, GRU

  - 真正意义上解决了长期依赖问题。LSTM和GRU通过引入门控机制，极大地缓解了RNN的梯度消失/爆炸问题，成为当时处理序列问题的标准模型。

序列到序列模型 (Seq2Seq)

- 序列到序列模型 (Seq2Seq) - 语言模型的“应用与扩展”
  - 核心思想: 由一个编码器（Encoder）RNN和一个解码器（Decoder）RNN组成。编码器将整个输入序列压缩成一个上下文向量（Context Vector），解码器则基于这个向量生成输出序列。注意力机制（Attention）的引入允许解码器在生成每个词时，动态地关注输入序列的不同部分，是革命性的突破
  - 代表: Seq2Seq

  - Seq2Seq本身是一个条件语言模型（在给定输入序列的条件下，生成输出序列）。它将语言模型的应用从单纯的文本生成/概率计算，扩展到了机器翻译、对话系统、文本摘要等更广泛的任务，是后续Transformer模型的重要思想来源。

预训练语言模型

预训练语言模型

- 基于Transformer的语言模型
  - 使用自注意力机制来捕捉序列内的依赖关系。模型可并行计算，有效捕捉长距离依赖
  - 代表Transformer，是现代所有大型语言模型（LLM）的架构基础
- 大规模预训练模型
  - 自编码模型 (Auto-Encoding Models)
    - 通过掩码语言模型任务进行预训练，能够同时看到上下文（双向），对自然语言理解任务（如分类、实体识别）特别有效。它本身更像一个强大的特征编码器
    - 代表: BERT
  - 自回归模型 (Auto-Regressive Models)
    - 严格遵循从左到右的生成方式，一次预测一个词。适合自然语言生成任务（如写作、对话）
    - 代表: GPT (Generative Pre-trained Transformer) 系列，通常所说的“语言模型”
  - 编码器-解码器模型 (Encoder-Decoder Models)
    - 结合BERT和GPT结构，拥有完整Transformer Encoder和Decoder。适合序列转换任务（如翻译）
    - 代表: T5, BART