[草稿] 备注 1: 引言和 Word2Vec

CS 224n: 深度学习中的自然语言处理

2023 年冬季

课程讲师: 克里斯托弗·曼宁, 约翰·休伊特

[2] 作者: 约翰·休伊特

johnhew@cs.stanford.edu

3 过去笔记的贡献者: Francois Chaubard, Michael Fang, Guillaume Genthial, Rohit Mundra, Richard Socher

摘要。本笔记简要介绍了自然语言处理（NLP）领域，然后讨论了 word2vec，并介绍了将词表示为从分布信号中学习到的低维实值向量的基本而美妙的想法。

1 介绍自然语言处理

自然语言处理是科学和工程的一个领域，专注于自动系统的研究与开发，这些系统能够理解并生成自然语言（即人类语言）。

1.1 人类和语言

人类语言是用于高效分享和存储复杂思想、事实和意图的交流工具。正如[Manning, 2022]所论述的，语言所支持的交流复杂性是物种间独一无二的人类智能。作为对智能系统的设计和研究感兴趣的科学家和工程师，人类语言对我们既是令人着迷的研究对象——毕竟，它已经进化到易于学习和实用的程度——也是在其他模态（例如，视觉）也感兴趣的环境中与人类交互的强大工具。

1.2 语言与机器

Human 儿童，与一个丰富的多模态世界和各种形式的反馈互动，以极高的样本效率（没有观察到太多的语言）和计算效率（大脑是高效的计算机器！）学会了语言。尽管在过去的几十年中，自然语言处理（NLP）取得了（令人印象深刻的！）进步，但我们仍然远远没有开发出具有儿童几分之一获取能力的学习机器。构建语言学习机器的一个基本（仍然相当开放）问题是表示问题；我们应该如何在计算机中表示语言，以便计算机能够稳健地处理和/或生成它？本课程的重点在于深度学习提供的工具，这是一个高度有效的工具包，用于表示自然语言的广泛多样性。

和它有时遵循的一些规则和结构。本课程的大部分内容将致力于这个问题的探讨，而本文的其余部分将讨论一个基本的子问题：我们如何表示单词？不过，在此之前，让我们简要讨论一下学习现代自然语言处理技术后你可以构建的一些应用。

1.3 几种 NLP 的应用

自然语言处理算法越来越有用并被部署，但它们的失败和局限性仍然很大程度上是不透明的，有时很难检测。以下是其中一些主要应用；这份列表旨在引起您的兴趣，而不是详尽无遗：

机器翻译。可能是自然语言处理最早且最成功的应用之一和驱动力，MT 系统学习在不同语言之间进行翻译，并在数字世界中无处不在。然而，这些系统对于世界上 7000 多种语言中的大多数来说仍然存在失败，长文本翻译的困难以及确保翻译语境正确性的问题使得这一领域仍然是富有成效的研究领域。

问答和信息检索。“问题”这个概念应该显得过于宽泛——我们不能将任何问题都表达为问答形式吗？——但在自然语言处理中，问答通常与信息查询问题相关联（“阿布扎比的埃米尔是谁？”、“我如何获得英国实习签证？”）。不断扩展可回答问题的范围，为答案提供来源，以及在互动对话中回答问题——这是研究领域中发展最快的方向之一。

问答和信息检索。“问题”这个概念应该显得过于宽泛——我们不能将任何问题都表达为问答形式吗？——但在自然语言处理中，问答通常与信息查询问题相关联（“阿布扎比的埃米尔是谁？”、“我如何获得英国实习签证？”）。不断扩展可回答问题的范围，为答案提供来源，以及在互动对话中回答问题——这是研究领域中发展最快的方向之一。

不断扩展可回答问题的范围，为答案提供来源，以及在互动对话中回答问题——这是研究领域中发展最快的方向之一。

文本的总结和分析。有无数的理由想要了解（1）人们在谈论什么和（2）他们对这些事情的看法。公司想要进行市场研究，政治家想要了解人们的观点，个人想要获得复杂主题的可消化总结。NLP 工具可以极大地增加信息的获取，同时也可能用于监视、企业和政府。在你继续前进并决定你要构建什么时，请记住“双重用途”这一方面。

注意：语音（或手语）转文本。自动将口头或手语语言（音频或视频）转录为文本表示是一个巨大的且有用的应用，但我们将在很大程度上

避免在这门课程中。部分原因是历史和方法论上的；原始信号处理方法和专业知识通常在其他课程（224s！）和其他研究社区中有所涵盖，尽管最近技术方面有一些趋同。

在自然语言处理的所有方面，现有的大多数工具只适用于世界上的少数几种语言（通常只有一种，最多可能有 100 种），并且在较少使用的和/或边缘化的方言、口音以及其他方面表现尤为糟糕。除此之外，近年来构建更好系统的成功已经远远超过了我们对这些系统的描述和审计能力。文本中编码的偏见，从种族到性别再到宗教以及其他方面，都会反映并往往被 NLP 系统放大。考虑到这些挑战和考虑因素，但怀着做好科学研究和建立值得信赖的系统以改善人们生活的愿望，让我们来看看 NLP 的第一个有趣问题。

2 词的表示

2.1 符号与所指

考虑句子

Zuko 为他的叔叔泡茶。

单词 Zuko 是一个符号，代表某个（现实或想象的世界中的）实体 Zuko。单词 tea 也是一个符号，指的是一个所指的事物——也许是某种特定的茶。如果改为说 Zuko 喜欢为他的叔叔泡茶，注意符号 Zuko 仍然指的是 Zuko，但 tea 现在指的是一个更广泛的类别——茶的总体，而不是某种特定的热茶。考虑以下两个句子：

Zuko 为他的叔叔做咖啡。
Zuko 为他的叔叔做饮料。

哪一句“更像”关于茶的句子？饮料可能是茶（也可能完全不同！），而咖啡肯定不是茶，但却是相似的，不是吗？Zuko 和叔叔相似是因为他们都是指特定类别的具体实例吗？

词的意义是无尽复杂的，源自人类交流和实现世界目标的意图。人们使用连续媒体——言语、手势，但产生的是离散的、符号化的结构——语言，来表达复杂的意义。表达和处理语言的细微差别和狂野性——同时实现强有力的信息传递——

人类交流和实现世界目标的意愿使得语言的形成成为一个无尽迷人的问题。让我们转向一些方法。

2.2 独立的词，独立的向量

什么是词？我不能为你定义一个词，但我可以给出一些

英语示例：茶，咖啡，缩写，干劲。单词反-

radate 我 hereby 定义 为 意味着 望眼欲穿地 看着 一个 不可食用 的 装饰， 希望 它 与 看起来 一样 可口。 如果 我 使用 这个 符号 与 其他人 交流 我 的 渴望， 那 对 我 来说 就 足够 成为 一个 单词 了。

也许 最 简单 的 表示 单词 的 方式 是 作为 独立 的 、 无关 的 实体。 你可能 会 想象 这 是 一个 集合，

{. . . , 茶, . . . , 咖啡, . . . , 抗皱剂}.

这里让我们介绍一些术语。我们将把词型视为有限词汇表中的一个元素，独立于实际在上下文中观察到该词。所以，我们刚刚写了一组类型。一个词元是类型的一个实例，例如，在某些上下文中观察到的。是词汇表中的一个元素；一个（词）词元是类型在上下文中的一个实例。

目前我们的词表示为每个词型提供了一个单一的表示，并且我们可能会在同一上下文中词元的任何出现使用相同的表示。

我们的词表示目前为每个词型提供了一个单一的表示，我们可能会在同一上下文中词元的任何出现使用相同的表示。

在本课程中，我们经常处理向量；独立分量的标准向量表示是 1-热或标准基向量集。因此，可能

$$v= \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \quad v_{咖啡} = \begin{bmatrix} \\ \vdots \\ 0 \\ 0 \\ 1 \\ \vdots \end{bmatrix} \tag{1}$$

where v= e, 第三个标准基向量, 和 v= e, 第 j 个标准基向量。

为什么我们将单词表示为向量？为了更好地进行计算。而在使用 1-hot 向量进行计算时，我们确实实现了不同的单词是不同的这一关键事实，但遗憾的是，我们没有编码任何有意义的相似性或其他关系。这是因为，例如，如果我们用点积作为相似性的概念（或 L2 距离，或 L1 距离，等等），我们计算的是：

$$vv= vv= 0, \tag{2}$$

所有单词彼此之间的相似性都是相同的。此外请注意，在图中，单词是没有顺序的，例如，按字母顺序排列——这是

一个重要的注释；这些字符串中没有（显式的）字符级信息，除了严格意义上的身份概念（这个词是否是另一词的字符/字节序列？如果是，它们有相同的向量；如果不是，它们有独立的向量）。由于当然不是所有词彼此都一样不相似，我们将转向一些替代方案。

2.3 带注释的离散属性的向量。

我们是否应该用特征和与语言类别及其他词的关系来表示词义，而不是用一热向量？

对于任何单词，比如跑者，我们都可以标注关于这个单词的大量信息。有语法信息，比如复数形式，还有派生信息，比如跑者是动词跑加上“执行者”或“主体”的概念（即跑步的人）。还有语义信息，比如跑者可能是人类、动物或实体的下位词。（下位词是指一种‘是’关系的成员；例如，跑者是

human.)

在英语和其他几种语言中，已经存在大量的标注信息资源。WordNet [Miller, 1995] 标注同义词、下位词和其他语义关系；UniMorph [Batsuren et al., 2022] 标注多种语言中的形态学（子词结构）信息。有了这些资源，可以构建出类似于

$$v= \begin{bmatrix} 0 \\ 0 \\ 1 \\ : \\ 1 \end{bmatrix} \begin{matrix} \\ \text{(第三人称单数动词) (饮料的下位词) . . .} \\ \\ \\ \text{(茶的同义词)} \end{matrix} \qquad (3)$$

(复数名词)

2023 年，这些方法产生的词向量并不是常态，也不会是这门课程的重点。一个主要的失败是，人工标注的资源在词汇量上总是比可以从自然文本中抽取词汇量的方法要少——更新这些资源的成本很高，而且它们总是不完整。另一个失败是维度性和嵌入有用性之间的权衡——要表示所有这些类别，需要一个非常高维度的向量（想想比词汇量大得多的维度），而现代基于密集向量的神经方法在这种向量上表现不佳。最后，在这门课程中，我们会看到一个持续的主题是，人类对什么是

一个主要的失败是，人类标注的资源在词汇量上总是比可以从自然文本中获取词汇的方法要少——更新这些资源成本高昂，它们总是不完整——文本应该具有的正确表示往往不如允许数据决定更多方面的方法表现得好——至少当有大量数据可供学习时是这样。

3 分布式语义和 Word2vec

深度学习的一个承诺是从数据中学习复杂对象的丰富表示。在自然语言处理中，一个相关的想法是，我们可以从数据中无监督地学习丰富的表示。无监督（或最近的，“自监督”）学习会从数据中提取信息，并尝试学习该数据中元素的属性，通常通过使用数据的一部分（可能是句子中的一个词）来尝试预测数据的其他部分（其他词）。在语言方面，这一想法早在多年前就被费尔 TH [费尔 TH, 1957] 捕捉得很好，他著名地说过

你通过一个词周围的词语来了解这个词的意思。

从高层次来看，你可以将茶周围的词语分布视为定义这个词意义的一种方式。

所以，茶周围出现了喝了、的、壶、水壶、包、美味、乌龙、热、蒸汽，等等。应该会变得清晰的是，与茶相似的词（如咖啡）会有相似的周围词语分布。虽然简单，但这在现代自然语言处理中是最有影响力和最成功的理念之一，它的类比已经广泛应用于各种学习相关领域。

分布假设：单词的意义可以从它出现的上下文分布中推导出来。
单词的意义可以从它出现的上下文分布中推导出来。

这就是高层次的概述。但像往常一样，细节很重要。一个单词离另一个单词有多近是什么意思？（紧挨着吗？两个位置之外吗？在同一个文档中吗？）如何表示和学习这种编码？让我们来看看一些选项。

3.1 共现矩阵和文档上下文

如果你被要求实现“用一个词周围的词的分布来表示这个词”的想法，你可能会立刻想到以下的想法：

1. 确定一个词汇表 V。

2. 创建一个大小为 |V| × |V| 的零矩阵。

3. 遍历一系列文档。对于每个文档，对于文档中的每个词 w，将文档中 w 出现的所有其他词的计数加到 w 对应行的 w 对应列中。

4. 将行归一化为和。

你刚刚为你的词汇生成了一个文档级共现矩阵！称这个矩阵为 X。矩阵 X 中的词嵌入 X∈ R（X 的一行）比我们以前的 1-hot ethat 要实质性地更加直接有用。

我们做出的一个决定是文档级共现。我们也可以这样说，一个词 w 只有在 w 出现得非常近，比如说在几个词以内时，才与 w 共现。这里有一个例子，其中几个相关的共现窗口被标记出来了：

更大的共现概念（例如，大的窗口或文档）会导致更语义或甚至主题编码的表示；更短的窗口会导致更语法编码的表示

[天气热，味道也好。[我倒了[茶 |{z} for]我叔叔].]

中心词

简而言之，较短的窗口（如上面标记为 1 的单个单词窗口）似乎编码了句法属性。例如，名词往往紧挨着或 is 出现。复数名词不会紧挨着 a 出现。较大的窗口倾向于编码更多的语义属性（在极端情况下，类似于主题属性）。注意如何 poured 或 delicious 可能离 tea 较远但仍相关。文档级别的窗口，对于大型文档（数千个单词），直觉上代表单词出现在哪些类型的文档中（体育、法律、医学等）。我们做出的另一个设计决策是用|V |-大小的向量表示显式的词频。这最终变成一个很大的错误。我们已经说过，高维向量在今天的神经系统中往往难以处理。但另一个问题是，原始词频会过度强调像 the 这样的常见词的重要性。取对数词频最终会更有用。 一篇非常有影响力的关于词表示的论文教会了我们更多关于原始共现方法的不足之处，通过引入 GloVe（Pennington et al., 2014），这是一种基于共现的词表示算法，其效果与我们在下一节中将要介绍的 word2vec 相当。然而，word2vec 的许多细节在我们后续课程中将要介绍的方法中仍然适用，所以我们将把时间集中在它上面。

3.2 Word2vec 模型和目标

word2vec 模型将每个固定词汇表中的词表示为低维向量（远小于词汇表大小）。它通过简单地函数学习每个词向量的值，该函数基于（通常是短的；2-4 个词）上下文中词的分布来预测。我们将描述的模型称为 skipgram word2vec 算法。

skipgram word2vec。如常，我们有一个有限的词汇表 V。令 C, O 为表示 C ∈ V（一个

中心词), O ∈ V (一个外部词，在中心词的上下文中出现)。我们将使用 c, o 来指代随机变量的具体值。令 U ∈ R, V ∈ R。请注意，V 中的每个词都与 U 的一行和 V 中的一个相关联；我们认为这是由于对 V 进行任意排序的结果。word2vec 模型是一个概率模型，如下所述，其中 u 代表 U 中与词 w ∈ V 对应的行（V 也同理）：

$$p(o|c) = \frac{\exp uv}{\sum \exp uv} \quad (4)$$

"这可能对你来说很熟悉，因为它就是 softmax 函数，该函数将任意分数（这里是从每个词汇的点积得到的分数）转换为概率分布，其中分数较高的事物获得较高的概率。请注意，给定中心词时所有词的概率向量 p(· | c) ∈ R 很像我们旧的共现矩阵 X 的一行。”

故事还没有结束；这只是个模型。我们如何估计参数 U, V 的值？我们学习最小化交叉熵损失目标，使其与真实分布 P(O | C) 相匹配：

$$\min_{U,V} \; E\left[-\log p(o \mid c) \right]. \quad (5)$$

这方程应该读作“在参数 U 和 V 的条件下，最小化由 O 和 V 的分布抽取的 o 和 c 的值的负对数概率的期望值”.

这里有很多丰富的细节可以深入探讨。我们如何执行最小化操作？我们如何“获取”随机变量 O 和 C？为什么使用负对数概率？为什么这种方法比共现计数更好？你能看出不是所有给定 c 的 o 的概率分布都能被这个模型表示吗？（这应该是好的吗？坏的吗？令人惊讶的吗？显然的吗？）现在，让我们来讨论一下如何在实践中实现这一点的几个细节。

3.3 从语料库估计 word2vec 模型

我们如何在实践中训练 word2vec？从我们上面给出的数学公式中，指定 word2vec 模型相对透明：构造矩阵 U 和 V，并可以写出概率的数学公式。然而，可能还不明显如何估计参数：（1）如何计算给定 U 和 V 值的方程 5 中的期望值，然后（2）如何执行最小化操作。让我们从 1 开始。

Word2vec 经验损失。设 D 为文档集合 {d}，其中每个文档是一个单词序列 w, . . . , w，所有 w ∈ V。设 k ∈ N 为正整数窗口大小。我们定义中心词随机变量 C 和外部词随机变量 O 与这个具体数据集的关系。O 取每个文档中每个单词 w 的值，对于每个这样的 w，外部词是 {w, . . . , w, w, . . . , w}。因此，我们的 Eqn 5 目标变为：

$$L(U, V) = \sum_{\in D} \sum_{i=1}^{m} \sum_{j=1}^{k} - \log p(w_{i-j} \mid w), \quad (6)$$

其中你将注意到我们对 (1) 所有文档中的 (2) 所有单词的 (3) 窗口中出现的所有单词的给定中心词的外部词的似然性求和。

现在，我们如何进行最小化?

基于梯度的估计 在高层次上，我们通过从相对无信息的猜测开始，逐步迭代地朝着局部最优改进猜测的方向移动，来尝试找到“好的”U 和 V，以满足我们所指定的目标。这是通过基于梯度的方法实现的，关于这些方法的完整描述超出了本文的范围。简而言之，标量函数 f 关于参数矩阵 U 的梯度（即：导数）∇f 表示为了最大化增加 f 的值，需要将 U 向哪个方向（局部）移动。因此，在实践中，我们像这样随机初始化 U 和 V：U, V ∼ N(0, 0.001)(从零均值、小方差的正态分布中独立抽样的矩阵)，然后进行一些迭代，执行以下过程：

$$U = U - \alpha \nabla L(U, V). \quad (7)$$

这应该被读作在第 i + 1 次迭代中设置 U 的值为前一次迭代中 U 的值，再加上一个在目标 L(U, V) 方向上局部最优改进 U 的小步长（α 小）.
随机梯度。这里还存在一个关键细节（除了如何计算梯度函数 ∇(·)，我们稍后会讨论这一点）：计算 L(U, V) 非常昂贵，因为它需要遍历整个数据集。我们不是精确计算目标函数，而是使用基于随机梯度的优化方法，在每一步的 Eqn 7 中使用少量样本来近似 L(U, V)。我们可能会这样做

而不是通过采样文档 d, . . . , d~ D 来精确计算目标函数

$$\hat{L}(U, V) = \sum_{d,\ldots,d} \sum_{=1}^{m} \sum_{=1}^{k} - \log p(w_{i-j} \mid w), \quad (8)$$

3.4 通过梯度计算。

一个词向量梯度步如何影响参数？让我们计算一下数学并建立一些直觉。特别是，我们将写出损失相对于中心词 c 的参数 v 的部分梯度。我们从写出梯度并让梯度运算符通过求和开始：

$$\nabla \hat{L}(U, V) = \sum_{i=1}^{m} \sum_{j=1}^{k} -\nabla \log p\left(w_{i-j}^{(d)} \mid w_i^{(d)}\right), \quad (9)$$

$$\sum_{\in D}$$

直观上，这些项的和的梯度就是它们梯度的和。我们来计算概率的梯度，为了简洁起见，我们将写 w

$$_{i-j}^{(d)}$$ 再次记作 o，和 w $$_{i}^{(d)}$$ 记作 c。

让我们先取和式的一项并用对数法则拆分它：

$$\nabla \log p(o \mid c) = \nabla \log \frac{\exp uv}{\sum uv} \quad (10)$$

$$= \nabla \log \exp uv - \nabla \log \sum_{=1}^{n} \exp uv \quad (11)$$

PartA　　PartB

A 部分。我们先求导 A 部分，因为它更容易。

$$\nabla \log \exp uv = \nabla uv \text{ 逆运算} \quad (12)$$

$$= u \text{ 为什么？} \quad (13)$$

为了理解最后一个等式（标记为“为什么？”）为什么成立，可以考虑向量 v 的每个单独维度。uv 的偏导数为 ∇uv = ∇∑uv = u。这是因为求和中的只有一个是 v 的函数，且 ∇uv = u 由单变量微积分得出。当我们把这些单变量导数堆叠在一起时，我们得到 [u , u] = u，如所写。梯度的形状是 u，而不是 u，这是约定；按照约定，我们将任何对象的梯度设置为其对象的形状，这可能涉及一些重塑。

Part B. 现在让我们区分 Part B。

$$
\begin{aligned}
-\nabla \log \sum_{x=1}^{n} \exp uv &= \frac{1}{\sum \exp uv} \nabla \sum_{x=1}^{n} \exp uv && \text{偏导数；链式法则} \\
&= \frac{1}{\sum \exp uv} \sum_{x=1}^{n} \nabla \exp uv && \nabla \text{exp uv 导数的线性} \\
&= \frac{1}{\sum \exp uv} \sum_{x=1}^{n} \exp(uv) \nabla uv && \text{指数函数的导数；链式法则} \\
&= \frac{1}{\sum \exp uv} \sum_{x=1}^{n} \exp(uv) u && \text{点积的导数}
\end{aligned}
$$

现在我们将使用这个来获得一些见解。让我们将部分 A 和部分 B 结合起来，
做一点代数运算：

$$
u - \underbrace{\frac{1}{\sum \exp uv} \sum_{x=1}^{n} \exp(uv) u}_{\text{将这个移到求和符号下}} = u - \sum_{x=1}^{n} \underbrace{\frac{\exp(uv)}{\sum \exp uv}}_{\text{这是 } p(x \mid c)} u
$$

$$
= u - \sum_{x=1}^{n} \underbrace{p(x \mid c)}_{\text{这是一个期望}} u
$$

$$
= u - E[u]
$$

= “observed” - “expected”，

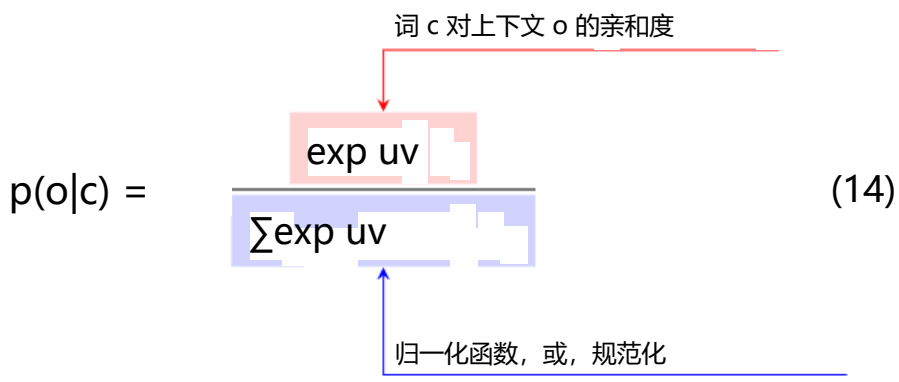
直观地讲，这一切都归结于这里最后一个方程；我们有实际观察到的单词向量：u。我们从那个向量中减去模型预期的向量——即模型分配给那个单词的概率总和乘以分配给那个单词的向量。因此，v 向量被更新为“更像”实际观察到的单词向量，而不是模型预期的单词向量。

如果你没有理解上面的数学推导，不要害怕；我建议你多看几遍，不要急于求成。如果你很快就理解了这一切而觉得无聊，那就利用你新获得的空闲时间去帮助别人学习吧！

3.5 Skipgram-negative-sampling

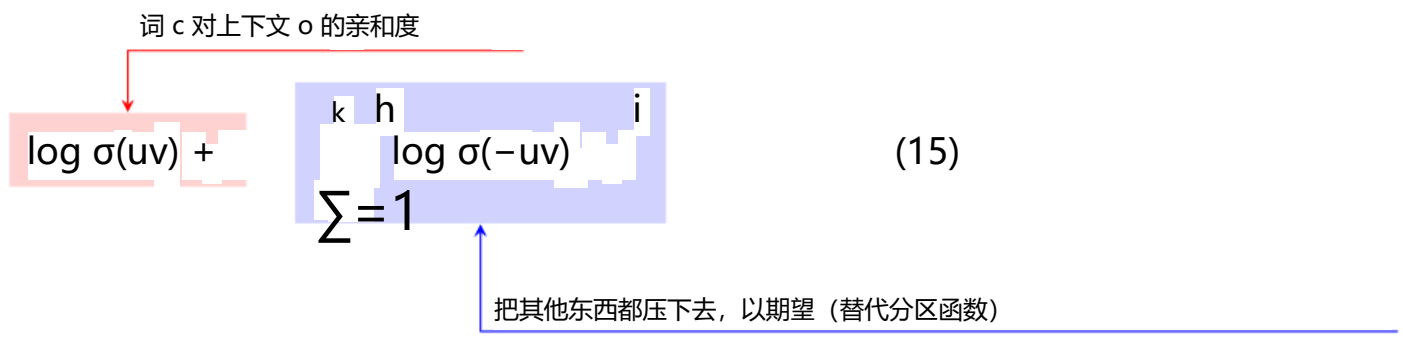
现在我们使用随机梯度估计，估计 word2vec 模型时的一个效率瓶颈是在计算\(\hat{L}(U, V) \)时计算精确的模型概率 \(- \log p(U, V)(o \mid c)\)。对于一个给定的词，计算未规范化分数 \(\exp(u^\top v)\) 很便宜。但是，计算分母中的分区函数（所有词的分数之和）很昂贵，因为它需要词汇表中每个词的一项。

直观上，分区函数做了什么，使得我们可以理解如何去除它？让我们再次写出 softmax：



从概率的角度来看，分区函数通过将分数归一化为总和为 1 来保证一个概率。（指数确保分数是非负的。）从学习的角度来看，分区函数“压低”除了观察到的词之外的所有词。换句话说，这个方程的分子鼓励模型使 u 更像 v；分母鼓励所有其他 w = o 的 u 更不像 v。负采样的直觉是，我们不需要一直压低所有的 u，因为大部分成本都来自那里。

然而，实际的带负采样的 skip-gram 目标（SGNS）会有所不同；我们在这里写出来：



其中，σ是逻辑函数，且 u~ p，这意味着 u 是从我们尚未定义的一个称为 p 的分布中抽取的。暂时可以这样理解：u 是从 V 的均匀分布中抽取的。这个目标函数在做什么？它有两个项，就像我们描述的原始 skipgram 一样。第一项鼓励 v 和 u 更相似，第二项鼓励 v 和 k 个随机词汇样本中的 u 更不相似。这里的直觉是，如果我们每次随机压下几个词，那么平均来说，这将类似于我们总是压下每个词的效果。

A 额外说明

A.1 连续词袋

A.2 奇异值分解

参考文献

[Batsuren et al., 2022] Batsuren, K., Goldman, O., Khalifa, S., Habash, N., Kiera´s, W., Bella, G., Leonard, B., Nicolai, G., Gorman, K., Ate, Y. G., Ryskina, M., Mielke, S., Budianskaya, E., El-Khaissi, C., Pimentel, T., Gasser, M., Lane, W. A., Raj, M., Coler, M., Samame, J. R. M., Camaiteri, D. S., Rojas, E. Z., López Francis, D., Oncevay, A., López Bautista, J., Villegas, G. C. S., Hennigen, L. T., Ek, A., Guriel, D., Dirix, P., Bernardy, J.-P., Scherbakov, A., Bayyr-ool, A., Anastasopoulos, A., Zariquiey, R., Sheifer, K., Ganieva, S., Cruz, H., Karahó ˇga, R., Markantonatou, S., Pavlidis, G., Plugaryov, M., Klyachko, E., Salehi, A., Angulo, C., Baxi, J., Krizhanovsky, A., Krizhanovskaya, N., Salesky, E., Vania, C., Ivanova, S., White, J., Maudslay, R. H., Valvoda, J., Zmigrod, R., Czarnowska, P., Nikkarinen, I., Salchak, A., Bhatt, B., Straughn, C., Liu, Z., Washington, J. N., Pinter, Y., Ataman, D., Wolinski, M., Suhardijanto, T., Yablonskaya, A., Stoehr, N., Dolatian, H., Nuriah, Z., Ratan, S., Tyers, F. M., Ponti, E. M., Aiton, G., Arora, A., Hatcher, R. J., Kumar, R., Young, J., Rodionova, D., Yemelina, A., Andrushko, T., Marchenko, I., Mashkovtseva, P., Serova, A., Prud' hommeaux, E., Nepomniashchaya, M., Giunchiglia, F., Chodroff, E., Hulden, M., Silfverberg, M., McCarthy, A. D., Yarowsky, D., Cotterell, R., Tsarfaty, R., and Vylomova, E. (2022). UniMorph 4.0: 通用形态学。在 Proceedings of 第十三次语言资源和评估会议，第 840-855 页，法国马赛。欧洲语言资源协会。

[Baum 和 Petrie, 1966] Baum, L. E. 和 Petrie, T. (1966). 有限状态马尔可夫链的概率函数的统计推断。数学统计年刊，37(6):1554–1563。

[Bengio et al., 2003] Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). 一种神经概率语言模型。J. Mach. Learn. Res., 3:1137–1155.

[Collobert et al., 2011] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. P. (2011). 从头开始的自然语言处理（几乎）。CoRR, abs/1103.0398。

[Firth, 1957] Firth, J. R. (1957). 应用普通语言学。哲学学会会刊, 56(1):1–14.

[Manning, 2022] Manning, C. D. (2022). Human Language Understanding & Reasoning. Daedalus, 151(2):127–138.

[Mikolov et al., 2013] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. CoRR, abs/1301.3781.

[米勒, 1995] 米勒, G. A. (1995). Wordnet: a lexical database for english. Communications of the ACM, 38(11):39–41.

[荣, 2014] 荣, X. (2014). word2vec 参数学习解释. CoRR, abs/1411.2738.

[Rumelhart et al., 1988] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1988). 神经计算: 研究基础. chapter 学习表示通过反向传播误差, pages 696–699. MIT Press, 美国马萨诸塞州剑桥.