作业 3 报告：基于 Word2Vec 模型的词向量有效性验证

吴頓
wd_0509@163.com

Abstract

本文主要工作流程包括预处理小说文本、训练 Word2Vec 模型和查找指定词语的相似词。它通过遍历文件夹中的每个小说文本，对其进行预处理并使用结巴分词进行分词处理，然后训练 Word2Vec 模型并查找指定词语的相似词。在处理过程中，会捕获错误并进行相应的处理。

Introduction

词向量是将单词表示为连续向量空间中的点，这种表示方式是自然语言处理中常用的一种技术。词向量的出现解决了传统的离散型表示方法（如独热编码）在处理自然语言时的一些问题，它将单词的语义信息编码到一个连续的向量空间中，使得具有相似语义的单词在向量空间中距离较近，从而更好地表达了单词之间的语义关系

词向量在自然语言处理中的应用非常广泛，主要体现在以下几个方面：

（1）词义表示： 词向量是将单词表示为连续向量空间中的点，其中相似的单词在向量空间中彼此靠近。这种表示方式能够捕捉到单词之间的语义相似性，例如在词嵌入空间中，词语"king"和"queen"之间的向量差异可能会与"man"和"woman"之间的向量差异相似。因此，词向量可以帮助模型更好地理解文本的语义。

（2）文本表示： 将文本表示为词向量的集合是自然语言处理中常见的操作。通过将文本中的每个单词映射到对应的词向量，并将这些词向量相加或连接，可以得到整个文本的向量表示。这种表示方式可以用于文本分类、文本聚类、信息检索等任务中。

（3）特征提取： 在传统的机器学习模型中，词向量通常被用作特征。例如，在情感分析任务中，可以使用词向量表示每个单词，并将这些词向量作为输入特征，然后应用逻辑回归、支持向量机等算法进行分类。

（4）语言模型： 词向量可以作为语言模型的输入。语言模型是一种用于预测文本序列的模型，可以用于自动文本生成、机器翻译等任务。通过将单词映射到对应的词向量，并将词向量作为输入，可以更好地捕捉到文本序列中的语义信息。

如何应用词向量：

（1）预训练模型： 可以使用已经训练好的词向量模型，如 Word2Vec、GloVe、FastText 等，直接在自己的任务中应用这些预训练模型得到的词向量。

（2）自定义模型： 如果预训练模型无法满足特定任务的需求，可以自己训练词向量模型。通常情况下，可以使用大量的文本语料库，如维基百科、新闻数据等，通过训练 Word2Vec、FastText 等模型来学习词向量。

（3）词向量可视化： 通过降维技术（如 t-SNE）将高维的词向量映射到二维或三维空间，可以将词向量可视化，以便更直观地理解单词之间的语义关系。

（4）词向量微调： 在特定任务中，有时候可以微调预训练的词向量，以使其更适应当前任务的特定语境。例如，在情感分析任务中，可以通过微调词向量来更好地捕捉到文本中的情感信息。

Methodology

Part 1:Word2Vec 模型

Word2Vec 模型是一种用于将单词映射到连续向量空间的神经网络模型，它是由 Google 的 Tomas Mikolov 在 2013 年提出的。Word2Vec 的核心思想是通过单词的上下文信息来学习单词的分布式表示，即将单词表示为高维空间中的向量，使得语义上相似的单词在向量空间中距离较近。
Word2Vec 模型有两种主要的训练算法：Skip-gram 和 CBOW（Continuous Bag of Words）。

Skip-gram 模型：

Skip-gram 模型的目标是根据给定的中心词来预测其周围的上下文词。具体地，它将一个中心词作为输入，并尝试预测在其周围窗口内可能出现的其他词的概率。训练过程中，Skip-gram 模型会通过最大化给定中心词下各个上下文词的条件概率来学习单词的向量表示。Skip-gram 模型适合于大型语料库和较少频繁词的任务，因为它能够捕捉到相对较远的上下文词之间的关系。

CBOW 模型：

CBOW 模型的目标与 Skip-gram 相反，它根据给定的周围上下文词来预测中心词。具体地，CBOW 将周围的多个词的向量作为输入，尝试预测中心词的向量。
训练过程中，CBOW 模型会通过最大化给定周围词下中心词的条件概率来学习单词的向量表示。CBOW 模型适用于小型语料库和频繁词的任务，因为它更容易捕捉到局部的语境信息。

Word2Vec 模型的训练过程通常使用随机梯度下降等优化算法，通过迭代更新模型参数来最大化目标函数（通常是似然函数），从而学习到单词的向量表示。训练完成后，可以使用学习到的单词向量进行各种自然语言处理任务，如词义推断、文本分类、信息检索等。Word2Vec 模型因其简单而有效的原理而成为自然语言处理领域中的经典模型之一。

Experimental Studies

Part 1:预处理文本数据

读取指定路径下的金庸小说文本文件，对文本进行预处理，包括去除特殊字符、分词等

操作，并将处理后的文本保存到指定的输出路径中。

名称	修改日期	类型	大小
白马啸西风	2024/6/2 14:47	文本文档	196 KB
碧血剑	2024/6/2 14:47	文本文档	1,402 KB
飞狐外传	2024/6/2 14:47	文本文档	1,262 KB
连城诀	2024/6/2 14:47	文本文档	651 KB
鹿鼎记	2024/6/2 14:47	文本文档	3,436 KB
绿色资源网	2024/6/2 14:47	Internet 快捷方式	1 KB
三十三剑客图	2024/6/2 14:46	文本文档	183 KB
射雕英雄传	2024/6/2 14:47	文本文档	2,588 KB
神雕侠侣	2024/6/2 14:47	文本文档	2,730 KB
书剑恩仇录	2024/6/2 14:46	文本文档	1,468 KB
天龙八部	2024/6/2 14:46	文本文档	3,424 KB
侠客行	2024/6/2 14:46	文本文档	1,039 KB
笑傲江湖	2024/6/2 14:47	文本文档	2,759 KB
雪山飞狐	2024/6/2 14:47	文本文档	382 KB
倚天屠龙记	2024/6/2 14:46	文本文档	2,747 KB
鸳鸯刀	2024/6/2 14:47	文本文档	102 KB
越女剑	2024/6/2 14:47	文本文档	47 KB

Figure 1：数据预处理输出文件

Part 2:模型训练

在 if __name__ == '__main__': 分支中，首先调用 read_novel 函数对数据进行预处理，然后进入模型训练的流程。在模型训练过程中，使用 Word2Vec 类构建词向量模型，传入预处理后的文本数据作为训练样本，设定了一些参数如窗口大小、向量维度等，然后训练模型。

模型参数的选择：

sentences: 训练模型所需的句子列表或可迭代对象，这里使用了 LineSentence 类，它从一个文件中逐行读取句子。

hs: 如果为 1，则使用 hierarchical softmax 训练模型；如果为 0（默认），则使用 negative sampling。

min_count: 忽略所有总频率低于此值的单词，默认为 5。

window: 当前词与预测词在一个句子中的最大距离，窗口大小是指当前词与目标词之间的最大距离，默认为 5。

vector_size: 输出的向量维度大小，默认为 100。

sg: 如果为 0，则使用 CBOW 模型进行训练；如果为 1，则使用 skip-gram 模型，默认为 0。

epochs: 训练的迭代次数，默认为 5。

```
try:
    model = Word2Vec(sentences=LineSentence(name), hs=1, min_count=10, window=7, vector_size=200, sg=0, epochs=200)
```

Figure 2：模型参数的选择

Part 2:相似词查询

训练好模型后，通过循环遍历预定义的一些关键词，如"张无忌"、"乔峰"等，利用训练好的模型查询这些词的相似词，并输出相似词及其相似度。

以下为部分词向量的相似词及相似度:

Table 1:杨过相似词及相似度

相似词	相似度
夫妻	0.23926986753940582
深深	0.22375965118408203
乌旺阿普	0.22108730673789978
武学之	0.2153572142124176
布袋中	0.2143290787935257
此外	0.21055401861667633
旁边	0.20969998836517334
藏经阁	0.20902714133262634
喷出	0.20799218118190765
云	0.20789237320423126

Table 2:少林相似词及相似度

相似词	相似度
山	0.38137170672416687
古往今来	0.25510475039482117
好端端	0.24523146450519562
既然	0.21425645053386688
茶道	0.20967116951942444
石破天	0.20847077667713165
叔	0.2080461084842682
帮会	0.20288856327533722
混蛋	0.1995907872915268
天大	0.19061347842216492

Conclusions

由 Table 1 与 Table 2 可知，与小说主人公名字相似的词与小说主题、主人公习惯用语、生活环境相关，部分相似词并不是完整的一个词语，猜测与 jieba 分词的分词效果相关。