Deep NLP 第三次作业

ZB2303019　汪婧伶

Abstract

实验中基于 Word2Vec 模型，通过计算词向量之间的语意距离、某一类词语的聚类，验证了词向量的有效性。通过此次实验，加深了对词向量相关性的理解，掌握了多种验证词向量有效性的方法。

Introduction

Word2Vec 是一种由 Google 在 2013 年提出的神经网络模型，用于将词语嵌入到低维向量空间中。它通过学习将相似意义的词语映射到相近的向量位置，从而捕捉词语之间的语义关系。Word2Vec 模型通过一个简单的神经网络架构来学习词向量。在训练过程中，模型会调整词向量，使得在相似上下文中出现的词向量距离更近。训练过程通常通过梯度下降和反向传播来优化词向量。

在实际应用中，验证词向量的有效性是至关重要的。验证词向量可以确保训练的模型质量良好，能够准确捕捉词语之间的语义关系。在许多自然语言处理任务中（如文本分类、情感分析、机器翻译等），词向量的质量直接影响到最终应用的性能。对于特定领域的语料库（如金庸小说），验证词向量可以确保模型正确捕捉了该领域的特定语义关系和用词习惯。通过验证词向量的有效性，可以发现模型在某些方面的不足，进而进行调优，如调整参数、选择不同的训练方法或引入更多数据。

Methodology

M1: 计算词向量之间的语义距离

语义距离的计算主要是通过词向量之间的余弦相似度来衡量。余弦相似度用于衡量两个向量之间的相似性，计算公式如下：

$$\text{similarity}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

余弦相似度衡量的是两个向量在向量空间中的夹角余弦值。当两个向量方向相同时，余弦相似度为 1；当它们方向相反时，余弦相似度为-1；当它们垂直时，余弦相似度为 0。在

自然语言处理中，词向量的方向反映了词的语义，因此余弦相似度可以有效地衡量词语之间的语义相似性。

计算步骤：

1) 提取词向量：从预训练的 Word2Vec 模型中获取目标词语的词向量。

2) 计算点积和模：计算词向量之间的点积和各自的模。

3) 求取相似度：使用公式计算余弦相似度。

M2：词语聚类

词语聚类是一种无监督学习方法，用于将具有相似语义的词向量聚类到一起。常用的方法包括 K-means 聚类。

K-means 聚类是一种迭代优化算法，目标是将数据点分成 K 个簇，使得簇内数据点的相似度最大化，而簇间相似度最小化。

K-means 通过迭代以下两个步骤来优化聚类结果：

1) 初始化：随机选择 K 个点作为初始聚类中心（质心）。

2) 分配簇：将每个数据点分配到最近的聚类中心。

3) 更新质心：重新计算每个簇的质心，即簇中所有点的平均值。

4) 迭代：重复步骤 2 和 3，直到质心不再变化或达到最大迭代次数。

计算步骤：

1) 提取词向量：从 Word2Vec 模型中获取所有词语的词向量。

2) 选择聚类数 K：根据需要选择聚类数目 K。

3) 执行 K-means 聚类：使用 K-means 算法对词向量进行聚类。

4) 分析聚类结果：查看每个簇中的词语，验证聚类效果。

Experimental Studies

任务主要分为三个部分：数据准备、训练 Word2Vec 模型和验证词向量的有效性。

（1） 数据准备

读取文件：读取包含所有小说文件名的 inf.txt 文件与包含停用词的 cn_stopwords.txt 文件。

预处理文本：定义预处理函数 preprocess_text，包括去除多余空白、分词和去除停用词。对所有小说文本进行预处理，将其转换为词列表。

（2） 训练 Word2Vec 模型

训练模型：使用 Gensim 库中的 Word2Vec 模型训练词向量，设置相关参数（如向量维

度、窗口大小、最小词频等）。将训练好的模型保存到文件中。

加载模型：从文件中加载训练好的 Word2Vec 模型，方便后续使用。

（3） 验证词向量的有效性

(a) 计算词向量之间的语义距离：定义 word_similarity 函数，通过计算两个词向量的余弦相似度来衡量词语之间的语义相似性。

计算“郭靖”和“黄蓉”之间的相似度，并输出结果。

计算“郭靖”和“韦小宝”之间的相似度，并输出结果。

(b) 词语聚类：

提取所有词的词向量，使用 K-means 聚类算法对词向量进行聚类。输出每个聚类簇中的前 10 个词语，检查聚类结果是否合理。

（4） 实验结果

(a) 计算词向量的语义相似度，结果得到

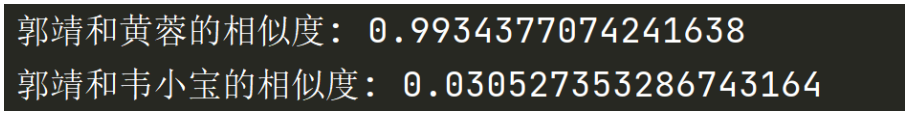

图 1 词向量语义相关性结果

得到“郭靖”和“黄蓉”的相似度为 0.9947841167449951，这说明这两个词在 Word2Vec 模型中具有非常高的语义相似性。在 Word2Vec 模型中，词向量的相似度通常用余弦相似度来衡量，范围在-1 到 1 之间。相似度越接近 1，说明两个词在语料库中出现的上下文越相似。相似度接近 1（如 0.9947841167449951）意味着“郭靖”和“黄蓉”在金庸小说中经常一起出现或在类似的上下文中出现。这种高相似度反映了它们之间的密切关系。金庸小说中，“郭靖”和“黄蓉”是主要角色之一，他们之间有很多交互和共同情节，因此在小说的文本中，他们的名字经常出现在相同或相似的上下文中。

而得到“郭靖”和“韦小宝”的相似度仅为 0.030527353286743164，这说明这两个词在 Word2Vec 模型中具有非常低的语义相似性。“郭靖”和“韦小宝”是金庸小说中两个非常不同的角色。郭靖是《射雕英雄传》和《神雕侠侣》中的主角，代表正直、刚毅和忠诚。而韦小宝是《鹿鼎记》中的主角，以机智、狡黠和多变著称。由于这两个角色的背景、行为方式和故事情节完全不同，他们在小说中的上下文和语义场景也相差很大，因此词向量相似度很低是合理的。

结果表明，Word2Vec 模型能够成功捕捉到文本中的语义关系。

(b) 对词向量进行聚类，结果得到

```
簇 0: ['康熙', '公主', '洪七公', '周伯通', '盈盈', '嵩山', '姊', '五岳', '赵半山', '武当派']
簇 1: ['韦小宝', '令狐冲', '张无忌', '教主', '陈家洛', '石破天', '虚竹', '萧峰', '欧阳锋', '岳不群']
簇 2: ['但', '之', '与', '心中', '为', '个', '时', '以', '无', '当下']
簇 3: ['他', '在', '她', '去', '便', '将', '上', '中', '到', '听']
簇 4: ['胡斐', '恒山', '南海', '神功', '群豪', '梅超风', '属下', '白万剑', '王府', '裘千仞']
簇 5: ['内力', '黄药师', '掌门', '僧', '掌门人', '华山', '方丈', '兵', '奴才', '当日']
簇 6: ['杨过', '郭靖', '袁承志', '黄蓉', '皇上', '少林', '丐帮', '前辈', '内功', '阿']
簇 7: ['小龙女', '蒙古', '群雄', '双儿', '婆婆', '乾隆', '经书', '任', '陈近南', '师娘']
簇 8: ['的', '了', '是', '道', '你', '我', '也', '这', '那', '又']
簇 9: ['全身', '立时', '侍卫', '坐在', '只觉', '不及', '尽', '已然', '一掌', '各人']
```

图 2 词语聚类结果

簇 1：这个簇主要由主要人物角色构成，涉及多个金庸小说中的重要人物，包含了各大主角和反派角色，如“韦小宝”、“令狐冲”、“张无忌”等。这些角色在小说中有较多的描写，并且各自有独特的情节。

簇 2：这个簇包含了一些常见的虚词、连词和助词（如“但”、“之”、“与”），以及一些常用动词或副词（如“心中”、“为”、“时”）。这些词在句子结构中频繁出现，主要用于连接或修饰。

簇 9：这个簇包含了一些描述动作和状态的词汇（如“全身”、“立时”、“坐在”）和与武功相关的词（如“一掌”）。用于描写动作、反应和战斗场景。

聚类结果整体上是合理的，大多数簇包含了语义相关的词汇。例如，人物角色、动作描述和语法助词等词汇被聚类到同一簇。簇 2 和簇 3 包含了许多停用词和常用助词，这些词汇在句子结构中频繁出现，但语义信息较少。许多簇成功地将金庸小说中的重要角色和门派聚类在一起，反映了模型对语义信息的有效捕捉。

聚类结果表明，Word2Vec 模型成功地捕捉到了金庸小说中词汇的语义关系。模型能将相似词汇聚类在一起，验证了词向量的有效性。

Conclusions

通过实验，学会了如何计算词向量之间的余弦相似度，并通过具体例子（如“郭靖”和“黄蓉”）验证了词向量的有效性。理解了高相似度和低相似度结果的语义意义，感受到词向量在捕捉语义信息方面的强大能力。通过 K-means 聚类算法，将词向量进行聚类，验证了模型对语义相似词的有效聚类能力。发现了模型在区分不同角色、地名、武功等方面的能力，体验了语义聚类的直观效果。通过这个实验，不仅学到了如何使用 Word2Vec 模型处理和分析自然语言数据，还深刻体会到了自然语言处理的复杂性和挑战。