Report of Deep Learning for Natural Langauge Processing

Weida Chen

908715799@qq.com

Abstract

本研究使用金庸的经典武侠小说作为语料库，使用 Word2Vec 模型训练词向量，并通过对词向量的聚类验证了模型生成词向量的有效性。

Introduction

一、 词向量

词向量是自然语言处理（NLP）中用于表示单词或短语的数值形式的技术。这些向量不仅仅是随机的数字集合，而是通过训练过程学习到的，可以捕捉到单词之间的各种语义和句法关系。词向量技术的主要目的是将单词转换成机器可以理解和处理的形式。词向量的类型主要有以下两种：

1. One-hot 编码

最简单的词向量形式。每个单词由一个很长的向量表示，向量的长度等于词汇表的大小，向量中只有一个元素是 1，其余都是 0。这个 1 的位置对应于该词在词汇表中的索引。这种表示法简单，但缺点是无法表达不同词之间的关系，且向量维度通常很大。

2. 分布式表示

Word2Vec 和 GloVe 等方法提供了基于单词的上下文的分布式表示。这些模型通过学习单词的使用环境来生成词向量，相似的单词会有相似的向量表示。这种表示法可以有效地捕捉语义信息，且向量的维度相对较小。

二、 Word2Vec 算法

Word2Vec 是一种广泛使用的词嵌入（word embedding）方法，由 Tomas Mikolov 等人在 2013 年提出。它的核心思想是将词语表示为高维空间中的向量，使得这些向量能够捕捉到词语之间的语义和句法关系。Word2Vec 通过神经网络模型学习词语的向量表示，主要有两种架构：

1. CBOW（Continuous Bag of Words，连续词袋模型）

CBOW 模型预测目标单词基于其上下文。具体来说，模型尝试预测给定上下文中的当前词。这个方法对小型数据集比较有效，可以更平滑地处理噪声数据。本次实验使用的模型为 CBOW 模型， 给定一个长度为 T 的文本序列，设时间步的词为 W(t),背景窗口大小为 m。则连续词袋模型的目标函数（损失函数）是由背景词生成任一中心词的概率。

$$\sum_{t=1}^{T} P\left(w^{(t)} | w^{(t-m)}, \ldots, w^{(t-1)}, w^{(t+1)}, \ldots, w^{(t+m)}\right)$$

2. Skip-gram（跳字模型）

与 CBOW 相反，Skip-gram 模型用当前词来预测其上下文。这种方法在大型数据集上表现更好，对低频词也能获得更好的表现。

两种模型都使用简单的一层或两层的神经网络，并通过优化一个逻辑回归目标函数（使用负采样或层次 softmax）来学习词向量。

三、 聚类算法

聚类是一种无监督学习技术，用于将数据点分组成多个集合，使得同一个集合内的数据点比其他集合的数据点更相似。聚类在各种领域中都有应用，如市场分析、图像分割、社交网络分析等。常见的聚类算法包括：

1. K-means 聚类:

这是最常用的聚类算法之一。算法首先随机选择 K 个中心，然后将每个点分配给最近的中心，形成 K 个簇。之后，重新计算每个簇的中心（即簇内所有点的均值），并重复此过程直到满足停止条件（如中心不再变化）。

2. 层次聚类:

层次聚类不需要预先指定簇的数目。它可以是凝聚的（从每个点作为单个簇开始，逐步合并最接近的簇）或分裂的（从所有点作为一个簇开始，逐步分裂至

更小的簇）。这种方法生成的是一个簇的层次树。

3. DBSCAN（Density-Based Spatial Clustering of Applications with Noise）：

DBSCAN 基于密度的聚类方法，可以识别任何形状的簇，并能有效处理噪声和异常点。它通过寻找高密度区域并扩展这些区域来形成簇。

Methodology

本实验首先对原始数据集进行处理，选取总数据集中的子集作为训练集。然后使用开源的 Gensim 库提供的接口来训练 Word2vec 模型。模型训练完毕之后，通过 Gensim 库中给出的接口函数，输出训练之后的模型，与某个给定输入词关联度最高的词或者是给定的某两个词之间的关联性。

M1：数据预处理

在读取语料后，首先去掉 txt 文本中一些无意义的广告、无关词语和标点符号等内容。此后利用 jieba 分词对语料进行分词，并读取停用词表进行文本过滤，最后将处理后的训练数据集重新保存进新的文件夹中。

```python
def get_corpus(file_path):
    corpus = ''
    # 去除无用字符
    r1 = u'[a-zA-Z0-9!"#$%&\'()*+,-./:;<=>?@，。?★、…【】《》？“”‘’！[\\]^_`{|}~「」『』（）]+'
    with open('../stopwords.txt', 'r', encoding='utf8') as f:
        stop_words = [word.strip('\n') for word in f.readlines()]
        f.close()
    # print(stop_words)
    with open(file_path, 'r', encoding='ANSI') as f:
        corpus = f.read()
        # 删除无用字符
        corpus = re.sub(r1, '', corpus)
        corpus = re.sub(r'\n|\u3000|本书来自免费小说下载站更多更新免费电子书请关注', '', corpus)
        f.close()
    # jieba分词
    words = list(jieba.cut(corpus))
    return [word for word in words if word not in stop_words]
```

图一 文本数据集预处理函数

M2：训练 Word2Vec 模型

Gensim 是一个用于自然语言处理的强大 Python 库，特别擅长处理主题建模和相似性检测任务。其中，Word2Vec 是 Gensim 库提供的一种工具，用于从文本数据中学习词向量。Word2Vec 模型基于神经网络，可以捕捉词语之间的复杂语

义和句法关系。本次实验直接使用 Gensim 库中提供的 Word2Vec 模型进行训练，并选取 6 本小说中的 6 位代表性人物，分析训练后与该人物和相关性最强的 5 个词。

M3：无监督聚类分析

为了进一步验证模型的有效性，使用 TSNE 将训练得到的模型中的词向量进行降维（方便展示效果），并使用 K-means 算法进行聚类。这里聚类用到的词为 6 本小说中的代表性人物。最终用散点图进行效果展示。

Experimental Studies

一、 相关词分析

在六本小说中各选取一名主角进行相关词语的分析，《射雕英雄传》中的郭靖，《神雕侠侣》中的杨过，《天龙八部》中的段誉，《笑傲江湖》中的令狐冲，《倚天屠龙记》中的张无忌以及《鹿鼎记》中的韦小宝，分析结果如下表所示。

表 1：各小说中代表性人物相关性

郭靖（射雕）		杨过（神雕）		段誉（天龙）		令狐冲（笑傲）		张无忌（倚天）		韦小宝（鹿鼎）	
黄药师	0.720	黄蓉	0.699	萧峰	0.625	岳不群	0.755	周芷若	0.720	康熙	0.721
黄蓉	0.718	小龙女	0.691	虚竹	0.624	林平之	0.687	赵敏	0.682	令狐冲	0.651
欧阳锋	0.688	李莫愁	0.669	慕容复	0.594	田伯光	0.656	张翠山	0.677	方怡	0.616
洪七公	0.682	周伯通	0.638	王语嫣	0.583	韦小宝	0.651	谢逊	0.661	太后	0.612
穆念慈	0.669	郭靖	0.635	乔峰	0.582	岳灵珊	0.642	金花婆	0.601	双儿	0.607

根据以上表格可以看出，词向量相似度较高的词在小说中也有一定关系。举例来说。

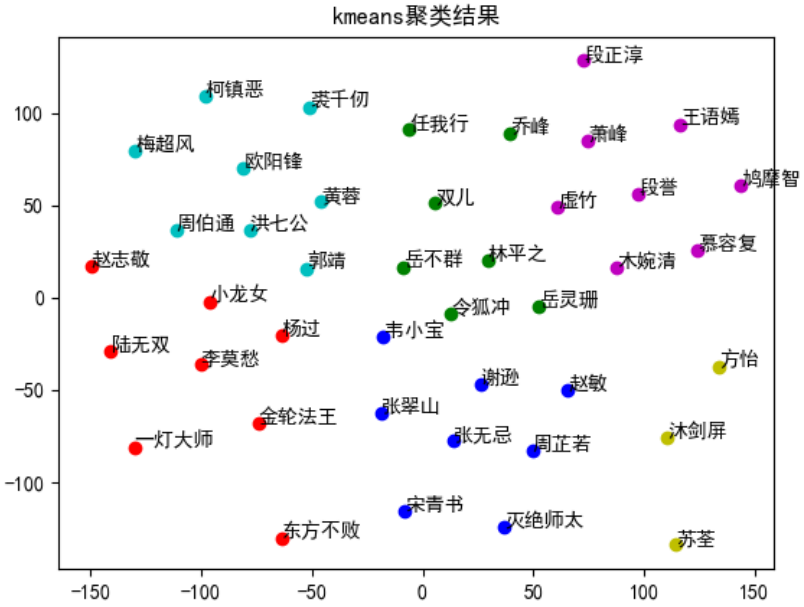
1. 射雕英雄传中，黄蓉是郭靖的妻子，黄药师是郭靖的岳父，欧阳锋是郭靖的仇人，洪七公是郭靖的师傅，穆念慈是郭靖的义妹。
2. 神雕侠侣中，黄蓉是杨过的义伯母，小龙女是杨过的师傅兼妻子，李莫愁是杨过的师伯，周伯通是杨过的大哥，郭靖是杨过的义伯父。

由此可见，基于 Word2Vec 模型得到的词向量之间的相关性在一定程度上能

够反映小说中人物角色之间的联系。

二、　聚类分析

本实验基于 Word2Vec 模型得到的词向量进行聚类（共分为六类），聚类结果如下图所示。可以看出，同一本小说中的人物基本被分到了同一类中，但也有少数划分错误的情况，例如双儿本属于《鹿鼎记》，却被分在了《笑傲江湖》中。



图二 K-means 聚类后人物关系图

Conclusions

本次实验基于金庸小说语料库使用 Word2Vec 模型进行词向量的训练，通过 K-means 聚类方法针对模型生成词向量的有效性进行了验证。实验结果显示，词向量之间的距离较小（相关性较大）通常表明这些词向量所对应的词在小说中具有较强的相关性，以此证明了模型生成词向量的有效性。

References

[1] https://blog.csdn.net/weixin_44966965/article/details/124732760