

# 基于深度神经网络的诗词检索<sup>1</sup>

梁健楠<sup>2</sup> 孙茂松 矣晓沅

(1. 清华大学计算机科学与技术系, 北京, 100084)

(2. 清华大学人工智能研究院, 北京, 100084)

(3. 清华大学智能技术与系统国家重点实验室, 北京, 100084)

liangjn16@mails.tsinghua.edu.cn

**摘要:** 中国古典诗歌是我国古典文学的代表之一, 是中华传统文化的宝藏, 源远流长。中国古典诗歌的研究是自然语言处理方向的一项重要且富有意义的工作。随着人工智能的发展, 神经网络在图像、文本等领域得到广泛的应用, 取得了显著的突破, 给人工智能与中国古典诗歌相结合提供了新的思路和方法。让机器去理解中国古典诗歌的韵律和意境是一件极具挑战的工作, 其中, 通过研究诗歌的相似性来提升机器对诗歌的理解这一研究课题被赋予了更为重要的意义。诗歌检索是对诗歌之间内容作对比, 查找出在语义和意境上相接近的诗歌, 这要求对整首诗歌的内容和意境有深入的理解。本文模型以数十万首古人诗作为基础, 利用循环神经网络(RNN)自动学习古诗句的语义表示, 并设计了多种方法自动计算两首诗之间关联性来计算两首诗歌之间的语义距离, 实现诗歌的推荐。自动评测和人工评测的实验结果都表明, 本文模型能够生成质量较好的诗歌检索结果。

**关键词** 神经网络 中国古典诗歌 诗词检索

## Neural Network-Based Poetry Retrieval

**Abstract:** Chinese classical poetry, with its long history, is one of the representatives of Chinese classical literature and a treasure of Chinese traditional culture. The study of Chinese classical poetry is an important and meaningful work in the field of natural language processing. With the development of artificial intelligence, artificial neural networks have been widely used in the fields of images and texts and have made significant breakthroughs, providing new ideas and methods for combining artificial intelligence with Chinese classical poetry. To make the machine understand the rhythm and artistic conception of Chinese classical poetry is a very challenging task, where the research topic of improving the understanding of poetry by studying the similarity of poetry has been given significance. Poetry retrieval is a comparison of the content between poetry, finding poems that are similar in semantics and artistic conception. This requires an in-depth understanding of the content and mood of the whole poem. Based on hundreds of thousands of ancient poems, this paper uses the recurrent neural network (RNN) to automatically learn the semantic representation of ancient poems, and designs a variety of methods to automatically calculate the correlation between two poems to further calculate the semantic distance between them, achieving the recommendation of poetry. The experimental results of automatic and manual evaluation show that the model can generate good quality poetry retrieval results.

**Keywords:** neural network; Chinese classical poetry; Poetry retrieval

## 1 引言

随着中国的崛起, 中国古典文化受到越来越多研究者关注。古诗词作为古典文化的代表之一, 深受人民的喜爱。

现在, 对中国古典诗歌自动生成的研究逐渐成为自然语言处理的研究热点。对诗词的研究最重要一步是机器自动语义理解。计算机的诗词自动语义理解需要理解内部词语的关联、意象的使用等规律, 从而促使人们对诗歌创作有更为直观的理解。诗词中的意象搭配、字词搭配等结果, 也可以辅助研究者对诗词展开进一步的文学研究。在当今人工智能火热的背景之下, 将传统文化和人工智能相结合, 能够为中华诗词的传承与发展贡献力量。近年来神经网络, 尤其是基于循环神经网络(Recurrent Neural Network, RNN)[1]的模型得到了极大的发展, 并且在图像识别、语音识别、机器翻译等任务上取得了显著突破, RNN 能够从大规模语料中学习得到句子的向量表示。前人的工作已经表明, RNN 能够用于英文诗歌[2][3]和中国古典诗歌[4][5][6]的自动生成, 并且能取得不错的效果。

<sup>1</sup> 基金项目: 国家社会科学基金 (13&ZD190)

<sup>2</sup> 作者简介: 作者梁健楠 (1991-), 男, 硕士生, 主要研究领域为自然语言处理。

诗词检索推荐是对诗词语义研究的一项重要任务，可应用在诗歌教学和诗歌领域的检索系统中。这要求推荐的诗词与原诗在语义上相近，意境上相似，风格上相仿，并要求创作者有丰富的诗词积累，对每首诗词有深入的理解。目前，大部分系统使用的检索推荐系统都是基于题目、诗人、标签信息做规则化推荐。

本文使用最新的深度神经网络方法学习诗词的句向量表示，用句向量来探索诗词检索推荐问题，实现出能够检索出与原诗在语义上相近，意境上相似，风格上相仿的诗词检索算法。句向量表示测量语义相似性，已在集句诗[7]中得到应用，取得不错的效果。

综上，文本的贡献如下：

1. 本文着重解决诗词检索任务，并设计了相应的自动评测实验；
2. 基于深度神经网络学习的诗词检索模型，该模型利用句向量可以自动计算语义，测量语义相似性；
3. 自动评测和人工评测结果都表明，对于查询诗词，本文模型能够检索出与其语义相关性强的诗词。

## 2 相关工作

对诗歌的自动分析和生成研究是计算机自动理解和使用人类语言的一个重要切入点。国内外，在这一领域的研究已经持续了数十年。国内相关研究起步于二十世纪九十年代。研究者在古诗文语料库建设、诗词检索、内容自动分析等方面做出了奠基性的工作。钱钟书先生亲自指导并参与筹建了中国社科院计算机室进行中国古籍数字化，运用计算机技术来保存、整理和运用中国古典文献。刘岩斌等[8]构建了一个古诗词研究系统，支持对古诗词电子文本的浏览、快速检索和统计。罗凤珠等[9]则构建了“倚声填词”格律检索与教学系统，支持对宋词词牌和词韵的检索；在古诗词分词这一任务上，俞士汶和胡俊峰[10]提出了基于“共现度”和“结合强度”的切分方法并构建了相应的古诗词典。苏劲松等[11]结合人工规则和统计信息对宋词进行了自动切分。此外，穗志方[12]结合知识库和互信息，实现了对宋代名家诗作的自动注音，在一定程度上解决了多音字带来的歧义问题。苏劲松[13]结合遗传算法和K最近邻方法实现了对宋词风格的自动识别，并采用多重松弛迭代算法进行了对宋词的情感分析。

随着技术的发展，研究者开始讲文本与高纬度的向量空间映射起来，来研究文本之间的关系。使用Word2Vector[14]获得词向量后，使用句子中词的词向量转换成句向量。2014年提出了一种基于循环神经网络的Seq2Seq模型[15]，使用了一个编码器(Encoder)加上一个解码器(Decoder)的结构，实现了一种端到端的机器翻译模型，该模型将输入的句子通过编码器得到隐向量。2018年，谷歌团队发布了BERT模型(Bidirectional Encoder Representations from Transformers, BERT)[16]。BERT是一种基于多层双向Transformer[17]编码器训练出来的语言模型，其性能对比许多任务中特定架构的系统都有所提升。在刚发布时，BERT刷新了11项NLP任务的最佳性能记录。

## 3 模型设计

### 3.1 任务及模型概述

目前，主流网站上设有诗词检索功能，满足用户拓展诗词的积累的需求和加深对诗词的理解。大部分诗词检索算法是基于对古诗的题目、作者和内容等信息做字符串匹配检索和基于人工标注的标签分类进行推荐。字符串匹配只是单纯地从字面上求近似，并没有理解到诗词的内容语义，结果并不一定准确。而人工标注需要的成本很高，标注的诗词数量有限。因此，这两种方法都有一定的局限性。

本研究提出使用基于深度神经网络的向量表示来进行诗词检索。首先，形式化地定义诗词检索问题：

设查询的诗词记为 $Q_{1:n} = Q_1 Q_2 \dots Q_n$ ， $n$ 为该诗词的句子数， $Q_i$ 为该诗词的第 $i$ 个句子。候选诗词 $P_{1:m} = P_1 P_2 \dots P_m$ 为诗库 $\{S\}$ 中的诗词， $m$ 为该候选诗词的句子数， $P_i$ 为该候选诗词的第 $i$ 句。

定义评分函数 $R(Q_{1:n}, P_{1:m})$ 来衡量候选诗词 $P_{1:m}$ 与查询诗词 $Q_{1:n}$ 的相关程度。最后根据评分函数 $R(Q_{1:n}, P_{1:m})$ 对诗库 $\{S\}$ 中所有候选诗词进行排序，给出推荐诗词。

本文涉及的诗词主要在近体诗范围内进行讨论。

本文把句向量表示拓展到诗词检索问题上。下面将介绍本研究在基于深度神经网络的文本向量表示来实现诗词检索的尝试。

### 3.2 句向量的选取方法

把文本映射成句向量方法有很多。下面，本文将介绍两个比较新的深度学习的方法，来计算句子的句向量。

第一种方法是使用基于 LSTM 带 Attention 机制的 Encoder-Decoder 框架求句向量。将诗句 S 输入到 LSTM Encoder 中，从输出的隐向量计算该句诗句的向量表示，公式为：

$$v(S) = \frac{1}{T} \sum_{t=1}^T S_t, \text{ 其中 } T \text{ 为句长。LSTM 能够学习到较好的诗句向量表示。}$$

第二种方法是使用 BERT 模型 (Bidirectional Encoder Representations from Transformers, BERT 求句向量。诗句经过 BERT 后可以获得多层输出，每一层输出包括输入的诗句每个字对应的隐向量，通过取高层的隐向量求得对应的句向量。具体实验当中，分别使用了由 Google 提供的预训练好的中文 BERT 模型参数 BERT-base -chinese 和在 BERT-base-chinese 的基础上使用古诗库的语料作为输入再训练获得的参数。使用古诗库语料再训练，是因为谷歌提供的 BERT-base-chinese 使用大量的中文语料库，其中古诗的语料占比非常少，因此希望通过再训练，使得 BERT 的模型参数对古诗语料更敏感一些。

### 3.3 诗词向量

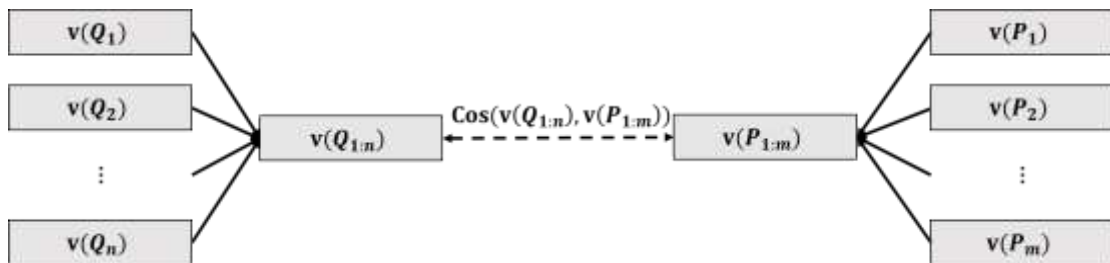


图 1: 诗词向量结构图

本文受由字的隐向量求句向量的启发，设计由句向量表示求出诗词的向量表示，如图 1 所示。一个简单的做法是将诗词  $P_{1:m}$  中的各句子对应的句向量做一次池化操作，这里使用的是均值池化处理。公式如下：

$$v(P_{1:m}) = \frac{1}{m} \sum_{i=1}^m v(P_i)$$

确定了诗词的向量表示后，可以使用向量的余弦距离来计算两首诗词的相似程度，称该方法为诗词向量诗词检索法 (Poetry Retrieval with Poem Vector, PRPVec)，公式如下：

$$R(Q_{1:n}, P_{1:m}) = \cos(v(Q_{1:n}), v(P_{1:m}))$$

该算法存在的问题是诗词向量糅合了整首诗词的字、句的向量，而忽略了诗词中字词的具体语义。

### 3.4 诗句向量

通过句向量直接取均值来作为诗词向量表示的方法过于简单，减少了诗词中的语义信息的关注。因此，本文保留诗词中的句向量表示，尝试利用合理的配对方法来计算两首诗词的相似程度。

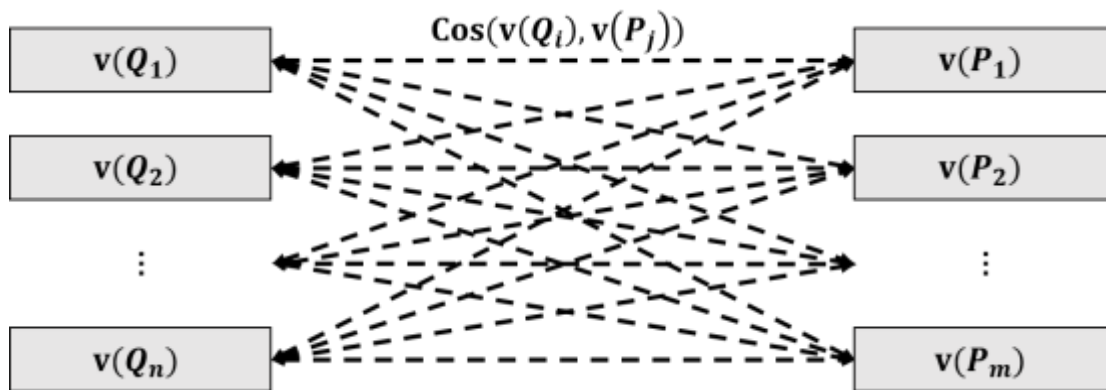


图 2：诗词检索的二分图结构

两首诗词 $Q_{1:n}$ 和 $P_{1:m}$ 的配对问题可看作是二分图，如图 2。诗句看做节点，左侧有 $n$ 个节点，分别为 $Q_1 Q_2 \dots Q_n$ 对应的句向量，右侧有 $m$ 个节点，分别为 $P_1 P_2 \dots P_m$ 对应的句向量。左侧第 $i$ 个节点连接右侧第 $j$ 个节点的边的权值为 $w(i, j)$ ，表示 $Q_i$ 与 $P_j$ 的相似程度，即 $w(i, j) = \cos(v(Q_i), v(P_j))$ 。以下介绍四种匹配的方法：

### 3.4.1 对应位置匹配

第一种匹配方法是考虑近体诗中一般每个位置的句子都有其任务，例如绝句中的四句诗句，一般都是按起承转合的顺序进行编排。开头两句诗一般描写风景，如“横看成岭侧成峰，远近高低各不同”；而第三句往往是感情的转折点——“不识庐山真面目”；最后一句“只缘身在此山中”用以抒发作者的观点和情绪。不同位置的诗句表示着不同的情景和情感，因此相同位置上的诗句，相似的可能性更高。

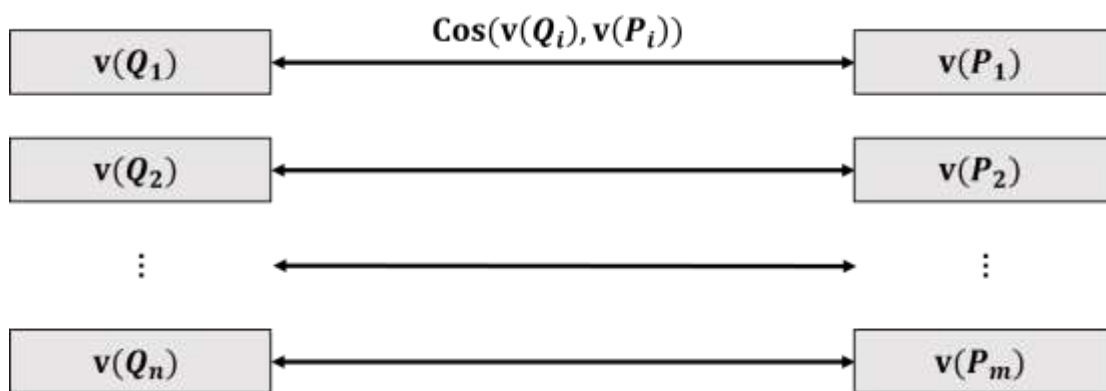


图 3：对应位置匹配法

如图 3 所示，按照句子所在的位置，求对应位置句子的句向量之间的余弦距离的均值，称该方法为行连接诗词检索法(Poetry Retrieval with Line Connection, PRLC)。公式如下：

$$R(Q_{1:n}, P_{1:m}) = \frac{1}{\min(n, m)} \sum_{i=1}^{\min(n, m)} \cos(v(Q_i), v(P_i))$$

固定了匹配的位置，算法可以使用较小的计算资源求得诗词间的相似程度。

本文做了一项数据分布统计，分别尝试了 8 首绝句和 8 首律诗作为查询诗词 $Q_{1:n}$ ，在查询诗词与所有候选诗词中，分析了查询诗词的每一句余弦距离最近的候选句位置的分布情况。结果见表 1 和表 2。

由表 1 中可以观察到，查询诗句的第一句在 27.78%的候选诗词（与查询诗词同是绝句的诗词）中余弦距离最近的是候选诗词的第一句，余弦距离最近的诗句为第二句、第三句和第四句的分别占 38.11%、5.16%和 28.95%。在表 1 和表 2 绝句和律诗的统计中，在大部分情

况下，查询诗句的诗句余弦距离最近的为相同位置的概率是最高的。但是，只有在绝句的第三句和律诗的第七句中余弦距离最近的为相同位置诗句的概率超过 1/2，其他情况都未达到一半。因此，行连接诗词检索法得到的结果在诗词相似性的表现上往往并不理想。

	第一句	第二句	第三句	第四句
第一句	27.78%	<b>38.11%</b>	5.16%	28.95%
第二句	22.72%	<b>46.18%</b>	3.01%	28.09%
第三句	13.04%	5.02%	<b>72.65%</b>	9.29%
第四句	16.15%	37.70%	4.12%	<b>42.03%</b>

表格 1：绝句各句最佳匹配句分布

	第一句	第二句	第三句	第四句	第五句	第六句	第七句	第八句
第一句	14.94%	<b>16.67%</b>	7.03%	13.96%	8.99%	14.53%	12.75%	11.13%
第二句	10.48%	<b>27.25%</b>	0.42%	17.09%	0.30%	23.54%	0.42%	20.50%
第三句	9.79%	0.99%	27.52%	0.58%	25.57%	0.52%	<b>34.07%</b>	0.97%
第四句	8.58%	17.26%	7.46%	<b>17.87%</b>	7.85%	13.63%	4.83%	22.51%
第五句	12.87%	0.72%	27.95%	0.53%	<b>32.87%</b>	0.38%	23.98%	0.69%
第六句	6.39%	<b>24.18%</b>	6.27%	12.94%	6.38%	23.14%	7.09%	13.61%
第七句	9.27%	1.25%	17.89%	0.78%	14.02%	0.55%	<b>53.02%</b>	3.22%
第八句	6.79%	16.84%	5.10%	16.55%	5.16%	14.02%	9.74%	<b>25.79%</b>

表格 2：律诗各句最佳匹配句分布

### 3.4.2 全连接匹配

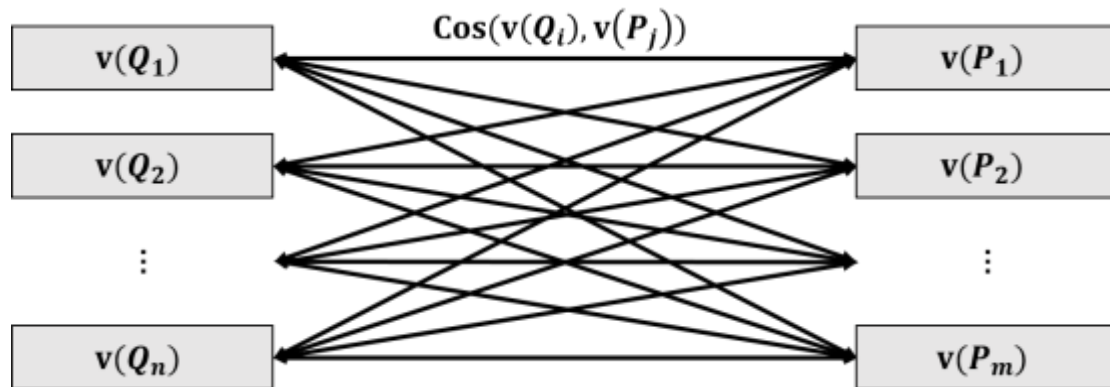


图 4：全连接匹配法

第二种匹配的方法是考虑到语义上最相近的句子，不一定在两首诗词的对应的位置。因此，本方法使用全连接匹配，求出两首诗词的句子两两之间的余弦距离的均值，称该方法为全连接诗词检索法(Poetry Retrieval with Full Connection, PRFC)。公式如下：

$$R(Q_{1:n}, P_{1:m}) = \frac{1}{n * m} \sum_{i=1}^n \sum_{j=1}^m \text{cos}(v(Q_i), v(P_j))$$

但是，该方法会导致余弦距离相对较远的形成噪音，使得两首诗词的评分函数  $R(Q_{1:n}, P_{1:m})$  得分降低，影响候选诗词  $P_{1:m}$  最终的排名。

### 3.4.3 最佳匹配

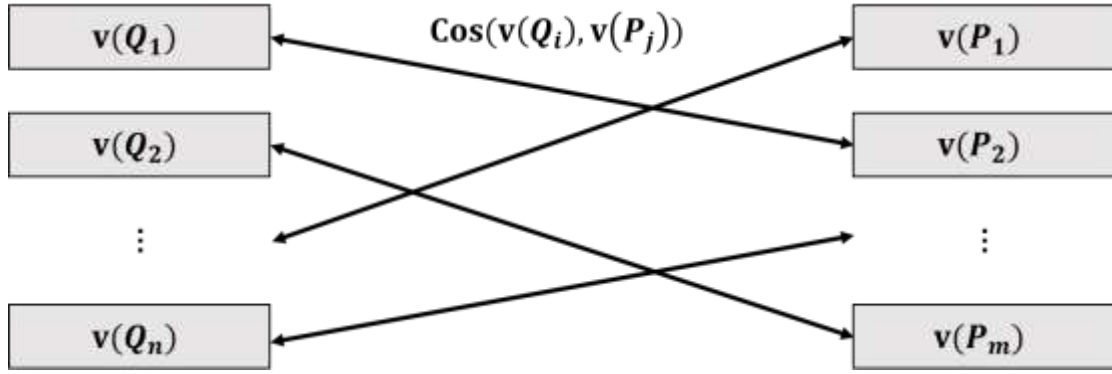


图 5: 二分图最佳匹配法

在实际检索中, 读者更加关注诗句的语义相近的句子情况。为了将注意力放到语义最相似的诗句对上, 本研究使用二分图最佳匹配来解决这一问题。设二元组集合  $\{(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)\}$ , 满足  $1 \leq k \leq \min(n, m)$ , 且  $\forall (i, j), 1 \leq i, j \leq k, i \neq j$ , 满足  $x_i \neq x_j, y_i \neq y_j$ , 称为二分图匹配  $A$ , 将所有满足以上条件的二分图匹配, 组成二分图匹配集, 记为  $\{A\}$ 。二分图最佳匹配即集合  $\{A\}$  中匹配边的权重和最大的匹配。本研究称该方法为最佳匹配诗词检索法 (Poetry Retrieval with Best Matching, PRBM)。公式如下:

$$R(Q_{1:n}, P_{1:m}) = \max_{A \in \{A\}} \frac{1}{|A|} \sum_{(i,j) \in A} \cos(v(Q_i), v(P_j))$$

二分图最佳匹配相比于上述其他方法, 最大的弊端是计算复杂度极高, 计算一对诗词的二分图最佳匹配使用暴力枚举, 需要的时间复杂度是  $O(N!)$ 。虽然使用 KM 算法计算二分图最佳匹配可以有效地提升计算效率, 但依然相对于之前的方法耗时更大, 难以投入到实际系统中。

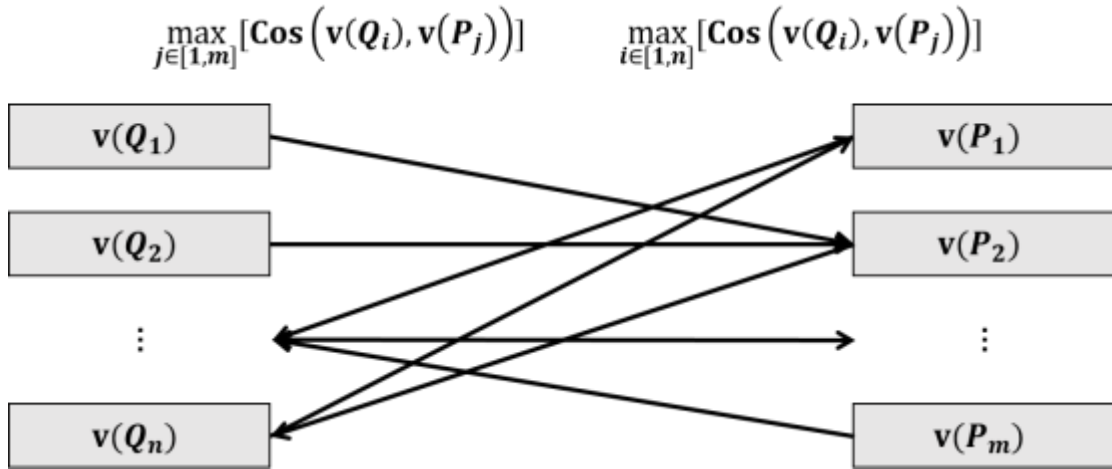


图 6: 行最佳匹配法

这里使用一种贪心的方式, 结构见图 6。在匹配的时候, 算法只需考虑每句话的局部最优, 计算  $Q_{1:n}$  和  $P_{1:m}$  中的每一句能匹配到的最优余弦距离的和, 称该方法为行最佳匹配诗词检索法 (Poetry Retrieval with Line Best Matching, PRLBM)。公式如下:

$$R(Q_{\{1:n\}}, P_{\{1:m\}}) = \frac{1}{n+m} \left[ \sum_{i=1}^n \max_{1 \leq j \leq m} \cos(v(Q_i), v(P_j)) + \sum_{j=1}^m \max_{1 \leq i \leq n} \cos(v(Q_i), v(P_j)) \right]$$

这样处理虽然会比使用二分图最佳匹配的方法多考虑一些连接, 但对质量的影响并不会很大, 而检索效率会比求二分图最佳匹配有显著的提升。

#### 4 实验分析

本文将对以上提出的五种诗词相似性的评分函数进行实验, 第一个实验通过人工评分来衡量不同评分函数的优劣, 第二个实验观察五个模型的检索效率, 第三个实验观察分析诗词检索推荐的实例。

以下是关于本文实验的说明：

- 诗词检索只考虑与查询的诗相同体裁的候选诗词，即查询的诗词是五言绝句，检索得出的诗词也会是五言绝句；

- 实验中引入基线实验——使用最长公共子序列长度作为得分，进行诗词检索推荐的排序标准，将用 PRLCS 表示。目前，部分网站的推荐算法使用的是基于最长公共子序列的相似性检索。

#### 4.1 人工评分

本实验随机挑选了故事库中 12 首近体诗，其中五言绝句 3 首，七言绝句 3 首，五言律诗 3 首和七言律诗 3 首，另外，使用诗歌生成系统生成了五言绝句、七言绝句各 2 首。对每一首古诗，使用 5 种方法分别计算出评分最高的 5 首诗，形成测试集。本实验中使用的诗句的向量表示，都是由生成模型中编码端输出的隐向量。

本实验邀请了多位志愿者参与人工测试，对上述每组诗的根据诗词之间的主题、语义和意境相关性对不同的方法进行排序。表 3 为各方法名次的平均数，数值越低，表示对应的方法检索出的诗词质量更好。

模型	五言绝句	七言绝句	五言律诗	七言律诗	总评
<b>PRLCS</b>	4.20	4.80	3.33	<b>1.33</b>	3.69
<b>PRPVec</b>	2.40	2.80	3.67	3.00	2.88
<b>PRLC</b>	5.40	4.00	5.00	4.33	4.69
<b>PRFC</b>	4.80	3.80	5.33	4.67	4.56
<b>PRBM</b>	2.40	<b>2.40</b>	2.33	3.67	2.63
<b>PRLBM</b>	<b>1.80</b>	3.20	<b>1.33</b>	4.00	<b>2.56</b>

表格 3：诗词检索人工评测结果

从实验结果可以看到，基于最佳匹配的方法 PRLBM 和 PRBM 在五言诗中都有较好的表现，其中 PRLBM 的质量最好。而在七言诗中，基于诗词向量的 PRPVec 表现稍显突出。比较有趣的是，基线方法 PRLCS 在七言律诗中效果最好，但其他体裁中质量却很不理想。导致这一项现象的原因可能是在长文本当中，最长公共子序列能匹配的字数更多，衡量近似程度的效果更好，而基于向量的表示方法则由于长文本的信息量比短文本大，造成的噪音较多，影响了最终结果的质量。

#### 4.2 效率分析

模型	运行时间(秒/首)				总评
	五言绝句	七言绝句	五言律诗	七言律诗	
<b>PRLCS</b>	3.40	37.32	53.82	155.67	62.55
<b>PRPVec</b>	<b>0.19</b>	<b>0.32</b>	<b>0.28</b>	<b>0.36</b>	<b>0.29</b>
<b>PRLC</b>	0.45	1.94	2.13	2.99	1.88
<b>PRFC</b>	0.90	5.01	10.79	16.63	8.33
<b>PRBM</b>	1.79	8.93	7170.84	11020.74	4550.57
<b>PRLBM</b>	1.17	6.27	13.57	20.91	10.48

表格 4：诗词检索效率分析

本实验测试不同模型在诗词检索过程中所需要花费的检索时间。实验所使用的诗库中有五言绝句 35,001 首、七言绝句有 180,877 首、五言律诗有 131,886 首和七言律诗 205,523 首。实验结果见表 4。本实验中使用的诗句的向量表示，都是由生成模型中编码端输出的隐向量。

其中，基于诗词向量的诗词检索法的检索速度最快，平均查询一首诗词的相关诗词可在 0.29 秒得到结果。基于最佳匹配的诗词检索法效率最低，查询一首诗词平均需要 4550.57 秒，接近一个半小时。

#### 4.3 BERT 与 Seq2Seq 对比

以上两个实验室基于生成模型中编码端输出的隐向量结果。以下，本文对比使用 BERT 模型输出的句向量和 Seq2Seq 的隐向量结果，在 PRLBM 方法下检索 Top-5 结果的效果。

本实验邀请了多位志愿者参与人工测试，对检索出来的 Top-5 结果的每首诗，根据诗词之间的主题、语义和意境相关性进行打分，最低分为 0，最高分为 5。

	五言绝句	七言绝句	五言律诗	七言律诗	总评
Seq2Seq	2.37	2.32	2.75	1.86	2.33
BERT	2.20	2.80	2.00	2.08	2.27

表格 5：对比结果

由表格 5 可以看出，整体来说 Seq2Seq 获得的句向量效果比 BERT 的句向量效果好。

#### 4.4 检索实例

本实验使用上文中介绍的 5 种方法进行诗词检索。将通过输入相同的诗词，检索得到评分最高的 5 首诗，进行对比分析。

这里，将《送杜少府之任蜀川》一诗送入检索系统，不同模型检索得到 Top-5 结果。以下是《送杜少府之任蜀川》：

城阙辅三秦，  
风烟望五津。  
与君离别意，  
同是宦游人。  
海内存知己，  
天涯若比邻。  
无为在歧路，  
儿女共沾巾。

实验结果见表 6。

模型	Top-5				
PRLCS	城阙辅三秦	残菊淮西路	西市多新鬼	白首对泷吏	风烟一以别
	风烟望五津	西风淹问津	南天少故人	骅汗愧逐臣	把酒恋同游
	与君离别意	三年同梦客	与君同应诏	溪山不改色	人去江楼晚
	同是宦游人	千里送归人	此别太惊神	风雨解留人	帆飞海国秋
	海内存知己	惭愧余知己	国论浮云变	坎壈知天意	青山驰远梦
	天涯若比邻	凄凉卜旧邻	封疆割肉匀	漂浮任此身	黄菊动离愁
	无为在歧路	怜君天下士	宁关儿女意	纷纷小儿女	曾是沧洲客
	儿女共沾巾	今在五湖滨	歧路泪沾巾	何必泪沾巾	能无念白鸥
PRPVec	城阙辅三秦	门外无游女	少年游上国	云海泛瓯闽	郎署飞符日
	风烟望五津	匡床祇自怜	名冠部儒绅	风潮泊岛滨	题书问谪居
	与君离别意	家园逢上巳	惜此三冬别	何知岁除夜	自因乡使到
	同是宦游人	客路类前年	依然万里人	得见故乡亲	翻觉旧交疏



	海内存知己 天涯若比邻 无为在歧路 儿女共沾巾	风雨花枝上 山川笠子边 因怀京洛士 谁送水衡钱	天涯同大被 终古见情亲 总是班行侣 明时献纳臣	余是乘槎客 君为失路人 平生复能几 一别十馀春	风月尘沙里 边风鼓角余 谁知迁客梦 夜夜绕鸾舆
<i>PRLC</i>	城阙辅三秦 风烟望五津 与君离别意 同是宦游人 海内存知己 天涯若比邻 无为在歧路 儿女共沾巾	又见秋风动 芦花江渚飞 忍看时序变 犹与老亲违 远信无他语 深情祇望归 应怜挥手日 儿女共牵衣	长至履微阳 江城百事荒 野天悬薄日 残叶堕浓霜 失路故人绝 入门新酒香 醉歌还起舞 儿女笑成行	曳杖青苔岸 系船枯柳根 德公方上冢 季路独留言 已占蒲鱼港 更开松菊园 从兹来往数 儿女自应门	垂老长为客 淹留媿此身 如何生白室 偏爱草玄人 腊酒倾三雅 春盘荐五辛 故乡千里外 儿女梦中亲
<i>PRFC</i>	城阙辅三秦 风烟望五津 与君离别意 同是宦游人 海内存知己 天涯若比邻 无为在歧路 儿女共沾巾	相望五千里 话别十三年 知我到江左 烦兄来日边 朋游俱健否 须发已卷然 共说儿童事 分明在目前	镜山欲相访 马首已之东 常日对衿佩 此行如燕鸿 将归逢所与 有约竟成空 若到斋中日 为予传主翁	水国无边际 舟行共使风 羨君从此去 朝夕见乡中 予亦离家久 南归恨不同 音书若有问 江上会相逢	少年游上国 名冠部儒绅 惜此三冬别 依然万里人 天涯同大被 终古见情亲 总是班行侣 明时献纳臣
<i>PRBM</i>	城阙辅三秦 风烟望五津 与君离别意 同是宦游人 海内存知己 天涯若比邻 无为在歧路 儿女共沾巾	海内故人少 市楼新酒醇 与君聊一醉 公袂此残春 北道邢台路 东州泗水滨 相期复何处 京洛旧风尘	征马顾还嘶 飞蓬东复西 伊予甘落落 吾子慎栖栖 海内谁胶漆 边陲正鼓鼙 相依傥严武 好住浣花溪	治乱亦无端 兴戎自晏安 会因千道设 岳拟万年看 城下盟初合 宫中戏未阑 可怜忠定泪 独洒北风寒	何处丈夫儿 不知名姓谁 弟兄三异相 上下两灵祠 江海为疆域 风雷听指麾 与人作方便 慷慨似生时
<i>PRLBM</i>	城阙辅三秦 风烟望五津 与君离别意 同是宦游人 海内存知己 天涯若比邻 无为在歧路 儿女共沾巾	海内故人少 市楼新酒醇 与君聊一醉 公袂此残春 北道邢台路 东州泗水滨 相期复何处 京洛旧风尘	过岭万馀里 旅游经此稀 相逢去家远 共说几时归 海上见花发 瘴中唯鸟飞 炎州望乡伴 自识北人衣	海外曾归客 春来复送人 相看同去住 临别且逡巡 去以初弦月 行当渐满轮 不应惟独照 凭尔辨疏亲	塞北曾相识 越南今又逢 如何两蓬鬓 都未歇萍踪 对酒心同感 思乡兴独浓 无能了公事 怅别意重重

表格 6: 诗词检索实例展示

可以看出,每个模型都检索到了原诗并将其排到第一位,在一定程度上验证了模型的正确性。从检索结果的第二首看,PRLBM及PRBM效果较好,其检索到的诗作在主题上与原诗一致,均为“送别友人”。两首诗在描写上和情感渲染上,也与原诗有相似之处。原诗先写景,由景及情,首先渲染了一片风烟迷茫的景色,然后表达了与友人同在仕途奔波的感慨,进而升华情感,展现出积极的心态和对友人的祝福。PRLBM和PRBM方法检索到的诗词有类似之处,起笔描写,渲染出暮春寥落之景,进而抒情,“相期复何处,京洛旧风尘”二句也体现出对未来的一种期冀。

相比之下,PRLCS方法检索结果的第二首虽然也为“送别”主题,但是整体抒情较多,景致渲染较少,并且情感偏向消极。而PRPVec和PRLC的结果主题为羁旅漂泊,与原诗不一致。PRFC的结果在情感上正面积积极,但内容是友人相逢而非送别,也与原诗不一致。PRLBM

与 PRBM 结果的前两首相同，但进一步对比第三首可以看出，PRBM 的结果主题为边塞思乡，风格与原诗相去较远。PRLBM 的结果主题为羁旅乡思，相比之下略优于 PRBM。从此例可看出，模型检索的效果整体与表 6 中的人工评测结果相仿。

## 5 结论与未来工作

本文探索使用句向量表示的诗词相似性评价。本文形式化地表述了所研究的问题，并提出了 5 种不同的评价方法来进行诗词检索。经实验，PRLBM 在诗词近似检索中有着不错的效果，且检索效率较高。本文工作已推出线上系统，可供用户使用。

未来的工作中，将优化诗句的向量表示。本文所使用的句向量表示为序列到序列模型的编码端和 BERT 模型的隐向量表示，是前沿的文本句向量表示的研究工作。此后，可以针对古诗词，设计更加符合古诗词特性的句向量表示。

## 参考文献

- [1] Boden, Mikael. "A guide to recurrent neural networks and backpropagation." the Dallas project (2002).
- [2] Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. Hafez: an interactive poetry generation system. In Proceedings of ACL 2017, System Demonstrations, pages 43–48. Association for Computational Linguistics, 2017.
- [3] Jack Hopkins and Douwe Kiela. Automatically generating rhythmic verse with neural networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pages 168–178. Association for Computational Linguistics, 2017.
- [4] Xingxing Zhang and Mirella Lapata. Chinese poetry generation with recurrent neural networks. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pages 670–680, Doha, Qatar, 2014.
- [5] Qixin Wang, Tianyi Luo, Dong Wang, and Chao Xing. Chinese song iambics generation with neural attention-based model. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, pages 2943–2949, New York, USA, 2016.
- [6] Xiaoyuan Yi, Ruoyu Li, and Maosong Sun. 2017. Generating chinese classical poems with rnn encoder-decoder. In Proceedings of the Sixteenth Chinese Computational Linguistics, pages 211–223, Nanjing, China.
- [7] 梁健楠, et al. "基于神经网络的集句诗自动生成." 中文信息学报 33.3: 126-135.
- [8] 刘岩斌, 俞士汶, 孙钦善. 古诗研究的计算机支持环境的实现[J]. 中文信息学报, 1997, 11(1): 27-36.
- [9] 罗凤珠, 李元萍, 曹伟政. 中国古代诗词格律自动检索与教学系统[J]. 中文信息学报, 1999, 13(1): 36-43.
- [10] 俞士汶, 胡俊峰. 唐宋诗之词汇自动分析及应用[J]. 语言暨语言学, 2002, 4.
- [11] 苏劲松, 周昌乐, 李翼鸿. 基于统计抽词和格律的全宋词切分语料库建立[博士学位论文]. 2007.
- [12] 穗志方, 俞士汶, 罗凤珠. 宋代名家诗自动注音研究及系统实现[J]. 中文信息学报, 1998, 12(2): 45-54.
- [13] 苏劲松. 全宋词语料库建设及其风格与情感分析的计算方法研究[J]. 厦门大学, 2007
- [14] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.
- [15] Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Proceedings of the 2015 International Conference on Learning Representations, San Diego, CA.
- [16] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
- [17] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems. 2017.

**梁健楠** (1991—), 男, 清华大学硕士生, 主要研究领域为自然语言处理。  
Email:liangjn16@mails.tsinghua.edu.cn