分 类 号：TP391　　密 级：无
学校代码：10414　　学 号：402017010869

江西师范大学

硕士研究生学位论文

基于深度学习技术的绝句生成方法研究

A Study of Chinese Quatrain Generation based on Deep Learning Methods

刘涵宇

院　　所：管理科学与工程研究中心　导师姓名：龚俊
　　软件学院
学科专业：管理科学与工程　研究方向：大数据与人工智能

二〇二〇年六月

独 创 性 声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：　　签字日期：　　年　　月　　日

学位论文版权使用授权书

本学位论文作者完全了解江西师范大学研究生院有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权江西师范大学研究生院可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名：　　导师签名：

签字日期：　　年　　月　　日　　签字日期：　　年　　月　　日

摘　要

诗歌是一种凝练而特别的文学形式，中国传统诗歌作为我国重要的文化遗产，体现了劳动人民非凡的智慧和创造力。绝句是中国传统诗歌中具有代表性的诗歌体裁，其在结构、平仄、押韵等方面都有严格的要求。创作一首合格的绝句对于普通人来说并不是件容易的事情，而对于计算机来说，如何自动生成绝句同样是充满挑战的课题。对绝句自动生成的研究，一方面可以降低诗歌创作门槛，让普通民众感受诗歌创作的魅力，有利于中华传统文化的传承；另一方面，绝句生成的研究让计算机进行文学创作成为可能，将给传统诗人及诗歌研究人员带来冲击，一定程度上将促进中国传统诗歌的创新与发展；同时绝句生成作为自然语言处理领域一项特别且有趣的研究，可启发其他文本类型的生成研究，促进自然语言处理相关技术的发展。因此对绝句生成的研究具有现实意义。

绝句等体裁诗歌生成的研究，经历了基于规则和模板的生成方法、基于统计机器学习的方法和基于深度学习的方法三个阶段，前两类方法生成的诗歌通常需要人工参与，且常出现较低级的错误，而随着深度学习技术的不断发展，基于深度学习的方法在诗歌生成中表现优异，成为了主流。本文在现有绝句生成方法基础上，针对绝句生成中主题漂移、语义不连贯等问题，提出了一种基于关键词转换扩展的绝句生成模型（Keyword Transformation and Expansion Quatrain Gene-ration Model，KTEQG），该模型将绝句生成分为关键词转换、关键词扩展和绝句生成三个阶段。首先对用户写作意图文本进行唯一关键词提取并进行关键词文言字词转换，然后基于转换后的主题关键词进行扩展，为每一句绝句分配主题关键词，最后基于注意力机制的编码器-解码器模型，将分配的主题关键词和历史生成的内容作为输入进行绝句生成。另外，本文针对现有的诗歌评估方法在切题、意境和情感等方面的不足，结合绝句的文学特点，完善了现有的人工评估方法，建立了更为科学的人工评估体系。

本研究与目前主流的绝句生成模型进行了对比实验，经过 BLEU 自动评估和人工评估，实验表明本文提出的 KTEQG 模型生成的绝句在格律、切题及内容表达上具有更为优秀的表现；模型还进行了图灵测试，实验表明 KTEQG 模型具有正常人类的智能，在绝句创作上达到了普通人类的创作水平。

关键词：自然语言处理；诗歌生成；注意力机制；编码器-解码器

Abstract

Poetry is a condensed and special literary form. Traditional Chinese poetry, as an important cultural heritage of our country, embodies the wisdom of our working people.The quatrain is a representative poetry genre in traditional Chinese poetry, which has strict requirements in terms of structure, flatness and rhyme. Creating a qualified quatrain is not an easy task for ordinary people, but for computers, how to automatically generate quatrains is also a challenging research topic.Research on the automatic generation of quatrains can lower the threshold of poetry creation, let ordinary people feel the charm of poetry creation, and help the inheritance of traditional Chinese culture; The impact of traditional poets and poetry researchers will, to a certain extent, promote the innovation and development of traditional Chinese poetry; meanwhile, quasi-sentence generation is a special and interesting research in the field of natural language processing, which can inspire other text types to generate research and promote Development of language processing related technologies. Therefore, the research on the generation of quatrains is scientific and feasible.

The research on the generation of quatrains has gone through three stages: the generation method based on rules and templates, the method based on statistical machine learning and the method based on deep learning. The poems generated by the first two methods usually require manual participation, and often have lower-level errors. With the continuous development of deep learning technology, deep learning-based methods have performed well in poem generation and become the mainstream.In this paper, based on the existing generation methods of quatrains, we propose a Keyword Transformation and Expansion Quatrain Generation Model (KTEQG) to solve the problems of topic drift and semantic incoherence in the generation of quatrains. The model divides the generation of quatrains into three stages: keyword conversion, keyword expansion and quatrains generation. First, extract the unique keywords of the user's writing intent text and perform keyword classical Chinese word conversion, then expand based on the converted topic keywords, assign topic keywords to each sentence, and finally based on the encoder-decoding of the attention mechanism The model uses the assigned topic keywords and historically generated content as input to generate quasi-sentences. In addition, this paper aims at the shortcomings of the current poetry evaluation methods in terms of topic, mood and emotion, and combines the literary characteristics of

quatrains to improve the existing manual evaluation methods and establish a more scientific manual evaluation system.

This study is a comparative experiment with the mainstream quatrains generation model. After BLEU automatic evaluation and manual evaluation, the experiment shows that the utterances generated by the KTEQG model proposed in this paper have better performance in metric, topic and content expression. The model was also subjected to a Turing test. The experiment showed that the KTEQG model has the intelligence of normal human beings, and has reached the level of ordinary humans in the creation of quatrains.

Key Words: Natural Language Processing; Poetry Generation; Attention Mechanism; Encoder-decoder.

目 录

1 绪　论

1.1 研究背景及意义

诗歌是一种简洁凝练而特别的文学形式，已有两千多年的历史，其起源可以追溯到上古时期的劳动号子、祭祀颂词等，而中国传统诗歌作为诗歌史上古老而特殊的一个分支，有其独特的魅力，是我国珍贵的文化遗产，体现了中华儿女非凡的智慧和创造力，具有极高的文学艺术价值。

中国传统诗歌具有严格的结构、平仄等要求，如图 1-1 为一首五言绝句，它由四行组成，每一行五个字，第一、第二和第四句最后一个字“晓”“鸟”“少”必须用同一个韵脚，同时每一句都有严格的平仄要求。普通人想要轻松创作一首合格的诗歌，并非易事，而如何让计算机学习诗人自动创作传统诗歌一直是自然语言处理领域非常具有挑战性的问题。

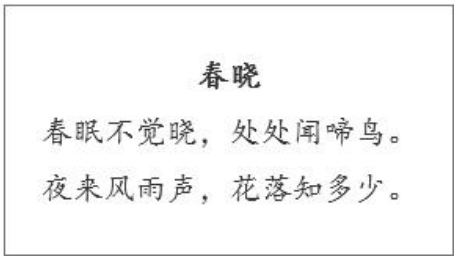

图 1-1 孟浩然五言绝句《春晓》

诗歌自动生成的相关研究最先在国外出现，而国内对于中国传统诗歌自动生成的相关研究直到 20 世纪 90 年代才开始，相对来说发展较慢，相比之下也存在许多不足，因此对于中国传统诗歌自动生成的相关研究还有许多进步空间。通过研究中国传统诗歌的自动生成，一方面可以降低诗歌创作门槛，让普通民众感受诗歌创作的魅力，有利于中华传统文化的传承；另一方面，诗歌生成的研究让计算机进行文学创作成为可能，将给传统诗人及诗歌研究人员带来冲击，一定程度上将促进中国传统诗歌的创新与发展；同时诗歌生成作为自然语言处理领域一项特别且有趣的研究，可启发其他文本类型的生成研究，促进自然语言处理相关技术的发展。因此对诗歌生成的研究具有现实意义。

21 世纪以来，随着计算机性能的提升及海量数据的出现，人工智能技术不断发展。2012 年，Hinton 和他的学生 Alex Krizhevsky 在 ImageNet 图像分类大赛中设计的 AlexNet 一举夺冠[1]，从此，深度学习开始被更多人熟知。深度学习是机器学习的一个分支，在图像识别，音频处理、自然语言处理等问题上，突破了

传统的机器学习瓶颈，取得了巨大成功，这也让深度学习开启了人工智能的新时代。自然语言处理技术（Natural Language Processing）通常包含语言学、数学、计算机科学等相关知识，是多门学科的融合，是人工智能领域一个重要的研究方向。自然语言处理技术是为了解决如何让人与计算机之间通过自然语言进行有效沟通这一问题，主要研究与之相关理论和方法，以实现人机有效交互。深度学习技术通过模仿人脑的神经机制，建立类似人脑结构的人工神经网络来解释计算机产生的数据，并通过底层特征组合的方式，来形成更为抽象的高层表示属性特征，从而达到发现数据的具体分布式特征表示。其在图像视觉、语音识别、自然语言处理等领域都表现突出，将深度学习技术应用到中国传统诗歌自动生成问题上，是自然语言处理领域一项特殊且具有挑战的尝试，具有很高的研究价值。

中国传统诗歌以唐诗最为闻名，唐诗中的绝句和律诗是唐朝诗歌最高水平的代表。绝句，也叫截句、绝诗，由四句组成，通常以五言绝句和七言绝句居多，有严格的格律要求[2]。据统计，唐诗约五万余首，而绝句这一种体裁就有一万余首，占约了其中五分之一，绝句作为唐诗重要的一部分，清朝学士王士祯曾经有过“唐代三百年以绝句擅场”的评论，从侧面反映出唐诗中绝句的地位，无论从数量还是质量上来讲，唐绝句都具有代表性，是绝句诗艺术成就的高峰[3]。而绝句这一诗歌体裁，也在唐朝形成了相对严格规范的章法要求，而更为规范的格律章法是有利于诗歌自动生成的研究的，因此本文选用绝句这一诗歌体裁进行诗歌自动生成方法研究，本文主要使用唐朝及唐之后的绝句作为训练语料，进行模型训练及实验。

本文在现有的绝句生成研究成果基础上，基于深度学习相关技术，针对绝句这一诗歌体裁进行诗歌生成方法研究，旨在通过本研究实现更好的绝句生成效果，生成达到人类创作水平的绝句诗歌。

1.2 研究现状

绝句是中国传统诗歌中具有代表性的诗歌体裁，对于绝句生成相关工作的研究，可看作诗歌生成相关工作的研究，因此本节对绝句生成的研究现状的阐述主要从诗歌生成研究现状进行论述。对于诗歌生成相关研究，在 20 世纪 60 年代的提出的 Word Salada（词语沙拉）[4]可以说是最早的诗歌生成方法，但该方法只是简单的将一些词语通过随机组合拼凑生成诗歌，如果严格来说，其生成的并不能称为合格的诗歌。而国内对于诗歌生成的研究直到 90 年代才开始，诗歌生成相关工作经过几十年的发展，有了许多尝试，并取得了一定的成果，诗歌生成的研究大致经历了基于规则和模板的生成方法、基于统计机器学习的方法和基于深度学习的方法三个阶段[5]。

基于规则和模板的生成方法主要有基于模板的方法[6][7][8]、基于实例推理的

方法[9]，这类方法更多的是通过模板设定进行填空组合生成，生成的诗歌很不连贯，甚至不能称作传统意义上的诗歌。在基于统计机器学习阶段，周昌乐等人在宋词生成中引入遗传算法，把宋词生成看作是最优化问题[10]；Yan 等人把诗歌生成问题看作指定写作意图的摘要生成问题，通过在诗歌语料库中检索词语进行排序然后进行组合形成诗句[11]；何晶等人把诗歌生成看作机器翻译问题进行诗歌生成，提出了一种基于统计机器翻译的中文诗歌生成模型[12]。该类方法常常依赖诗词方面的专业知识，必须要有相关专家先对人工规则进行设定，从而保证对生成诗词的平仄押韵等方面的约束，因此，针对不同体裁、语言的诗歌，就必须重新设计特定的人工规则，这使得该类方法迁移能力差，应用场景单一，缺乏通用性，生成的诗歌通常语义不连贯，是较低质量的诗歌生成。

随着深度学习技术的发展，基于深度学习技术的诗歌生成方法开始出现，而诗歌生成的研究也进入了新的阶段。基于循环神经网络的中国传统诗歌生成（RNN-based Poem Generator，RNNPG）[13]，通过用户提供关键词，再扩展关键字，给定相关格律限制，然后选择排名最好的作为诗的第一行，逐行生成整首诗歌。Wang 等人将基于注意力机制的神经网络机器翻译方法（Attention based Neural Machine Translation Network，ANMT）[14]应用到了宋词的生成中，将宋词生成看做翻译问题，将已生成的诗句作为输入，要生成的诗句当作目标输出进行诗歌生成。iPoet 是一种基于编码器-解码器框架的诗歌生成系统[15]，其通过模仿人类写诗再进行修改润色这一过程，创新的加入了诗歌打磨机制，可对生成的诗歌进行一次或多次的迭代打磨润色。Hafez 是一个可根据用户提供的主题生成任意数量诗歌的程序[16]，该程序基于编码器-解码器框架实现了对英文诗歌的生成。基于规划的诗歌生成模型（Planing based Poetry Generation,PPG）[17]模仿人类写诗列提纲这一举动，把写诗过程分成了诗歌规划和诗歌生成两阶段，通过规划子主题进行诗歌生成，该方法一定程度解决了主题漂移问题，但其基于现代白话文的主题规划，容易造成语义不良问题。清华大学自然语言处理与社会人文计算实验室提出了一种基于显著性上下文机制（Salient-Clue Mechanism）的诗歌生成方法[18]，通过已生成诗句提取具有显著特征的字词，并限制生成诗歌类型来进行诗歌生成，该方法具有很好的上下文连贯性，整体效果较好。但由于限制了诗歌生成类型可能造成流畅性问题且主题表达受到限制。基于以上讨论，我们发现现有基于深度学习技术的绝句生成方法仍有很大的进步空间。

1.3 主要研究工作

绝句的生成研究是自然语言处理领域一个特殊且具有挑战性的课题，本文在现有方法研究的基础上，提出了基于关键词转换扩展的绝句生成模型（Keyword Transformation and Expansion Quatrain Generation Model，KTEQG），专注于

绝句这一具有代表性诗歌体裁生成方法研究。本文主要工作内容如下：

（1）通过阅读大量相关文献，从基于规则和模板的生成方法、基于统计机器学习的方法和基于深度学习神经网络的方法三个阶段对现有绝句等体裁诗歌生成方法进行研究梳理，对现阶段绝句生成方法进行了系统全面的总结。

（2）本文基于绝句语料进行实验研究，收集整理了大量绝句诗数据，基于绝句的格律特征等特点，对训练语料进行了细致针对性的筛选和预处理。

（3）针对现有绝句生成模型普遍存在的主题漂移和语义不连贯等问题，首次提出了基于关键词转换拓展的绝句生成模型（Keyword Transformation and Expansion Quatrain Generation Model，KTEQG），在实际绝句生成中，取得了很好的生成效果。

（4）本文针对现有的诗歌生成模型人工评估方法在切题、意境和情感等方面评估指标欠缺的问题，结合中国传统诗歌的文学特点，完善了现有的人工评估方法，建立了更加科学的诗歌人工评估体系。

（5）从自动评估和人工评估两方面对绝句生成模型进行了实验评估，并通过图灵测试，试验表明本文提出的绝句生成模型在格律、切题及内容表达上具有更为的优秀表现，生成绝句达到了人类的创作水平。

1.4 组织结构

本文分五个章节，安排如下：

第一章，绪论。本章首先介绍绝句生成研究的相关背景及意义，并阐述现阶段绝句生成的国内外研究现状，然后介绍本文主要研究工作和组织结构安排。

第二章，基于关键词转换扩展的绝句生成模型设计。本章首先对基于深度学习的绝句生成问题进行介绍，然后对现阶段绝句生成相关工作进行研究讨论；接着提出基于关键词转换扩展的绝句生成模型（Keyword Transformation and Expansion Quatrain Generation Model，KTEQG），从关键词转换、关键词扩展、绝句生成三方面详细介绍模型的具体设计与实现。最后针对本研究中提出的绝句生成的相关评价体系进行介绍。

第三章，绝句生成模型算法的实施。本章主要介绍本文提出的 KTEQG 算法实施的相关研究：首先对绝句生成中绝句文本的表示方法进行研究，接着对模型神经网络的选择进行讨论，最后针对实验当中的优化算法进行讨论选择。

第四章，实验与分析。本章首先介绍绝句生成的数据集的选择与预处理，然后介绍了模型的训练，接着对本文提出的 KTEQG 模型进行实验与分析，通过自动评估、人工评估及图灵测试，来验证模型的先进性。

第五章，总结与展望。本章对论文的主要研究工作进行总结，并对未来的可能存在的研究方向进行讨论。

2 基于关键词转换扩展的绝句生成模型设计

2.1 绝句生成问题描述

本文绝句生成的研究基于深度学习技术，其整体流程如图 2-1。

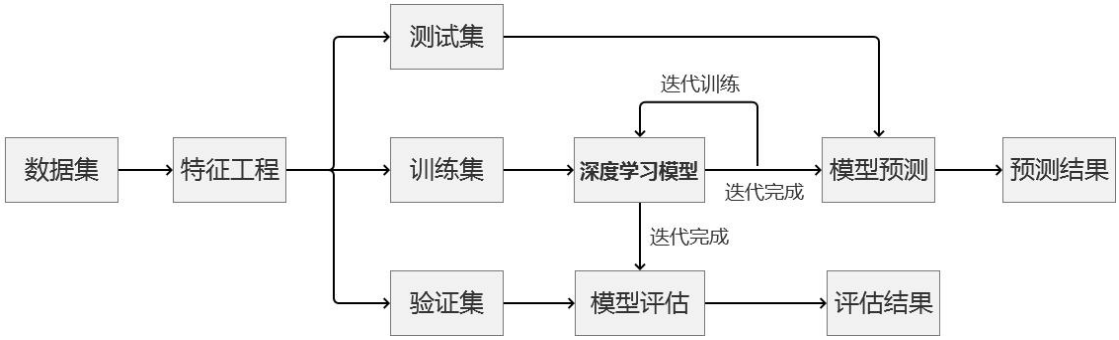

图 2-1：基于深度学习技术的绝句生成整体流程图

在绝句生成的研究中，首先对绝句数据集进行收集整理；特征工程阶段对绝句数据进行文本表示，接着将收集的绝句数据预处理，让计算机可以更好的理解处理绝句数据；然后将数据分为测试集、训练集和验证集，使用训练集对深度学习模型进行迭代训练，得到绝句生成最优模型；验证集用于验证模型的性能；测试集用于对训练完成的绝句生成模型进行测试评价。

2.2 相关工作

绝句是中国传统诗歌中具有代表性的诗歌体裁，对于绝句生成相关工作的研究，可看作是诗歌生成相关工作的研究，因此本节对基于深度学习技术的绝句生成相关工作的阐述主要从诗歌生成相关工作进行阐述。20 世纪 60 年代，国外便出现了诗歌生成的相关研究，而国内对于相关研究相对起步较晚，直到 20 世纪 90 年代才出现中文诗歌的生成研究，在将近 60 年的不断研究探索过程中，涌现了许多方法，诗歌生成的研究大致经历了基于规则和模板的生成方法、基于统计机器学习的方法和基于深度学习的方法三个阶段。

基于规则和模板的生成方法阶段主要有基于模板的方法[6][7][8]、基于实例推理的方法[9]，这类方法更多的是通过模板设定进行填空组合生成，生成的诗歌很不连贯，甚至不能称作传统意义上的诗歌。在基于统计机器学习阶段，周昌乐等人在宋词生成中引入遗传算法，把宋词生成看作是最优化 问题[10]；Yan 等人把

诗歌生成问题看作指定写作意图的摘要生成问题，通过在诗歌语料库中检索词语进行排序然后进行组合形成诗句[11]；何晶等人把诗歌生成看作机器翻译问题进行诗歌生成，提出了一种基于统计机器翻译的中文诗歌生成模型[12]。该类方法常常依赖诗词方面的专业知识，必须要有相关专家先对人工规则进行设定，从而保证对生成诗词的平仄押韵等方面的约束，因此，针对不同体裁、语言的诗歌，就必须重新设计特定的人工规则，这使得该类方法迁移能力差，应用场景单一，缺乏通用性，生成的诗歌通常语义不连贯，是较低质量的诗歌生成。

随着深度学习技术的不断发展，越来越多的研究者开始尝试通过深度学习技术进行诗歌生成，也让诗歌生成的研究也进入了新的阶段。

RNNPG（RNN-based Poem Generator）是一种基于循环神经网络的中国传统诗歌生成方法[13]，该方法通过用户提供关键词(例如，塞外，江南等)，且其关键词仅限于《诗学含英》[19]诗歌短语分类法中证实的关键词，然后生成器扩展这些关键字变成一组相关短语。在诗歌相关结构平仄的限制下，通过模型对句子打分，然后选择最高分句子作为诗歌首句，再由第一句生成第二句，一二句生成第三句，循环直至完成。该方法通过学习单个字符及其组合表示，自然地获取结构、语义和一致性约束，通过考虑到多句子的上下文信息来进行诗歌生成。与传统的诗歌生成方法相比，RNNPG 模型无需人工设计评估函数，可自动从诗歌语料库中进行学习，具有更好的诗歌生成效果。但由于诗歌生成主题关键词只应用于第一行诗句的生成，后续行的生成是根据已生成诗歌进行的，往后续诗句生成过程中，容易偏离诗歌主题，造成主题漂移问题。

基于注意力机制的神经网络机器翻译方法（Attention based Neural Machine Translation Network，ANMT）[20]主要运用于语言翻译，与传统的统计机器翻译不同，该方法建立一个单一的神经网络，可以联合调整，以最大程度地提高翻译性能，利用这种新的方法，其在英语到法语翻译的任务中取得了很好的翻译效果，可与现有的最先进的基于短语的翻译方法相媲美。而 Wang 等人[14]将该方法应用到了宋词的生成中，其将宋词生成当做翻译问题，把之前生成的诗句作为输入，下一要生成的诗句当作目标输出，通过双向 LSTM 模型对输入关键字进行编码，然后用基于注意力的 LSTM 模型解码得到诗句概率分布。模型生成过程中，需先提供第一句诗作为输入，然后将第一句作为输入，依次生成下一句，直到完成整首诗。该模型通过宋词语料库训练，进行宋词生成，宋词不同于大多数中国古典诗歌类型，涉及灵活的结构和复杂的节奏模式，因此还加入了一个基于 LSTM 的向量初始化解码器来区分词牌类型，宋词之前从未由机器成功生成，该模型可以很好地学习远距离的复杂结构和节奏模式。与 RNN 相比，LSTM 模型能够学习远程信息，学习更长的诗歌，让模型基于已生成的更远的诗歌进行建模，从而减少梯度消失问题，让语义更加连贯。此外，采用基于注意力机制的方法来提供

细粒度的监督，可以进行准确的字符级监督，从而模拟严格的结构规则和宋词的微妙情感状态。但是，该方法进行诗歌生成时须提供第一句诗，也就是说生成的诗歌的写作意图（主题）仅能体现在第一句诗歌上，即使 LSTM 模型可以学习更远的信息，但是随着诗歌行数不断增加，后面还是无法避免主题漂移的问题。

iPoet 是一种通过模仿人类创作诗歌反复修改打磨这一举动而建立的诗歌生成系统，其基于编码器-解码器框架，是一种具有打磨机制的神经网络诗歌生成方法[15]。该系统模仿人类进行文字创作后通常会再修改打磨这一举动，创新的加入了诗歌打磨阶段，对生成诗歌进行一次或多次的迭代润色。系统对写作意图进行编码，并根据写作意图解码生成诗歌是一个分层的概念，通过打磨将诗歌单遍生成扩展到多遍生成。该系统由意图表达、顺序生成、润色打磨三个部分组成，通过基于编码器-解码器的 RNN 诗歌生成和诗歌润色打磨的机制，将单遍生成扩展到多遍生成，从而得到了更好的诗歌生成效果，但该方法把之前生成的诗歌内容表示为固定长度的向量，因此在诗歌生成中就可能出现和历史生成的诗句失去联系的情况，从而导致诗句前后不连贯，造成主题漂移问题。

Hafez 是一个可以根据用户提供的主题生成任意数量诗歌的程序[16]，该程序基于编码器-解码器框架进行英文诗歌生成。系统首先选择一个特定的词汇库，计算每个单词的重音模式，根据用户提供的主题，计算出与主题相关的候选单词。然后度量候选词与主题词关联性，通过建立有限状态接受器（FSA）[16]，并为每个可能的符合节奏约束的单词序列提供一条路径，选择成对的满足约束条件的押韵词来作为诗句最后一个词。在进行诗歌生成时，作者首先尝试了 RNNLM 方法[21]从诗句右边开始往左边使用束搜索解码生成。然后使用基于注意力机制编码器-解码器方法，把每一句诗右边词语按顺序组成序列将该诗歌作为输出进行训练生成。通过评估基于注意力机制编码器-解码器方法具有更好的效果，与主题具有更好的关联性。Hafez 系统在生成诗歌过程中，通过保持诗句最后一个词语和主题相关，一定程度解决了主题漂移情况，因为其也将历史生成诗歌表示为固定长度向量，因此生成诗句语义连贯性不好，同时该系统应用于英文诗歌生成，针对中国传统诗歌特殊的规则、音律、平仄规则不一定适用。

王哲等人受人类写诗过程会先列出大纲这一举动启发，提出了基于规划的诗歌生成模型（Planing based Poetry Generation,PPG）[17]该方法把诗歌生成任务分成了诗歌规划与诗歌生成两阶段：诗歌规划阶段，用户首先输入写作意图（句子段落或关键词），诗歌规划模型从写作意图中提取多个关键词，为每一句诗歌分配一个主题词，如果用户写作意图输入文本序列太短无法提取足够多的关键词，则进行关键词扩展，获取足够子主题作为诗歌写作的大纲。诗歌生成阶段，基于编码器—解码器模型，根据诗歌规划阶段提供的关键词序列作为写作大纲逐行生成诗歌，生成过程中将每行关键词和之前的历史生成作为输入，生成下一行诗歌，

不断循环直至完成诗歌生成。PPG 模型通过给诗歌分配子主题，在解决主题漂移这一问题上取得了一定的效果，但该方法在主题规划过程中，提取的关键词常常出现现代白话文词语，而基于白话文关键词的诗歌生成，容易造成生成的诗句中也白话文，使得生成的诗歌失去了简洁精炼的特点而不像传统的中国古典诗歌，造成诗句语义不畅的问题。

清华大学自然语言处理与社会人文计算实验室提出了一种基于显著性上下文机制（Salient-Clue Mechanism）的诗歌生成方法[18]，并将该方法实现为“九歌”智能诗歌写作系统，取得了不错的诗歌生成效果。与先前尝试利用所有上下文的模型不同，该模型从先前已生成的诗句中选择了几个具有显著特征的字或词，为生成后续行提供了线索，作为输入来指导下一句诗歌生成，以此来保持整首诗的连贯性，同时加入用户意图和诗歌风格限制，以控制生成过程，从而进一步增强一致性。该模型通过人工标注语料，通过有监督的方式将诗歌风格限定在边塞诗，闺怨诗，山水田园诗三种诗歌风格，基于注意力机制的编码器-解码器模型进行诗歌生成，生成的诗歌具有很好的上下文连贯性，整体效果较好。但由于该模型仅限于三类诗歌的语料，只考虑了需要的类型诗歌部分而忽略语义不太好部分，因此造成流畅性问题。而生成诗歌类型的限制也使得主题表达受限制，因此该方法同样具有改进空间。

2.3 基于关键词转换扩展的绝句生成模型设计

由上述绝句生成相关工作的研究中，我们可以发现绝句等体裁诗歌生成中普遍面临两个问题：（1）主题漂移。即绝句生成过程中，随着生成诗句长度的增加，后续生成的诗句出现偏离写作主题的现象；（2）语义不连贯。即生成的诗句存在语义不连贯的问题，如诗句中出现白话文、语句不通等现象。本研究针对上述问题，在现有绝句生成方法的基础上，提出了基于关键词转换扩展的绝句诗生成模型（Keyword Transformation and Expansion Quatrain Generation Model，KTEQG），本模型首先通过用户输入的写作意图提取唯一主题关键词，然后将关键词进行文言文转换处理，再将转换后的主题关键词进行主题扩展，为每一行诗句分配相关主题词，基于注意力机制的编码器-解码器模型，将每一句关键词和历史生成作为输入进行绝句生成。

本模型分为关键词转换、关键词扩展和绝句生成三个阶段，其框架如图 2-2 所示。关键词转换阶段，首先对用户写作意图（用户输入的主题词、句子或文本段落）进行关键词提取，确定唯一主题关键词，然后将唯一关键词转换成文言文关键字词。关键词扩展阶段，基于绝句语料库将文言文关键词扩展至四个子主题关键字词，分配给绝句的每一行。绝句生成阶段，将关键词扩展阶段生成的子主题及历史生成的诗句作为输入，输出下一句诗，不断重复直到完成整首诗歌生成。

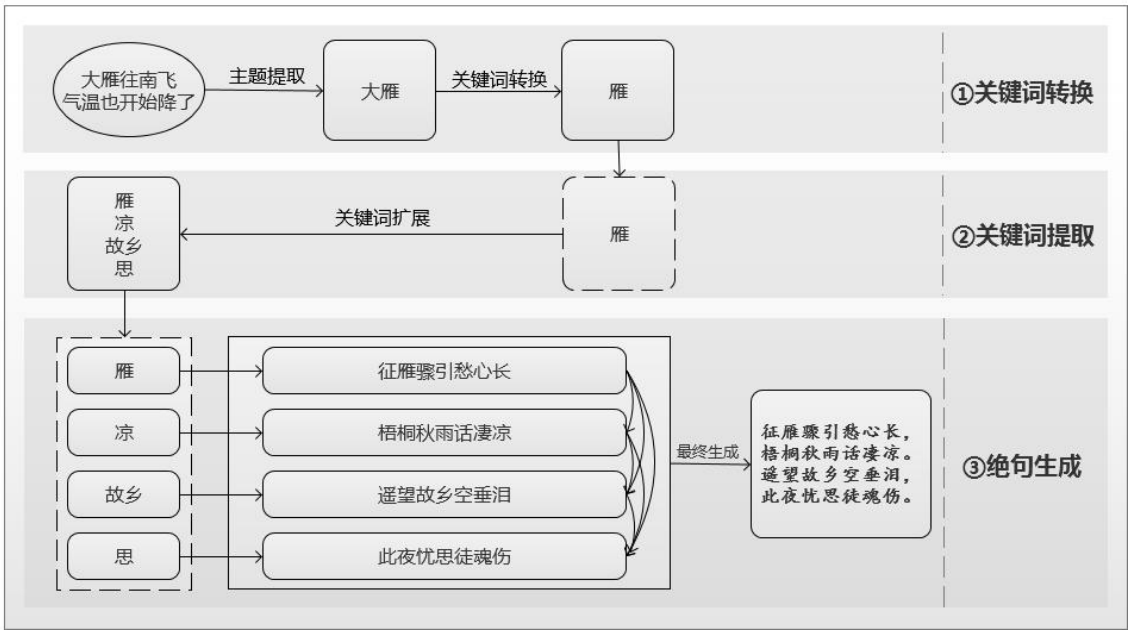

图 2-2：基于关键词转换扩展的绝句生成模型框架图

2.3.1 关键词转换

在关键词转换阶段，用户可输入任意文本序列作为写作意图，该文本序列可以是一个主题词、一个句子或是一段话。在以往的诗歌生成中[17]，关键词提取阶段将从用户输入文本序列中提取出多个关键词，使得关键词数量与诗歌总行数相同，若不够再进行关键词扩展，达到关键词与诗歌总行数相等。该模型在提取多个主题词时，易造成写作意图表达不明确的问题，如在 PPG 模型中当输入句子“春天像一位姑娘，踏着轻盈的脚步来了”，经过关键词提取，将提取出“春天”、“姑娘”两个关键词，再经过主题词扩展，生成诗歌。这句话主题关键词应该是“春天”，但因为模型选择多个主题词，“姑娘”和“春天”相关性低，因此容易造成主题偏离问题，同时提取的主题词过于白话文，不利于后续诗歌生成。

而本文提出的 KTEQG 绝句生成模型在用户写作意图关键词（主题词）提取时，为了明确写作主题，只提取评分最高的唯一关键词。因确定的唯一关键词容易出现白话文词语，其与诗词预料库中的古文词语不匹配，不利于子主题词生成和诗歌的生成，因此确定唯一关键词后，还将进行文言文词语转换，确定唯一文言文主题关键字词。下面将对主题关键词提取及转换的具体实现进行介绍。

关键词提取主要目标是从文本中自动提取出可以表示文本内容的词语。关键词提取可分为有监督、半监督和无监督的方法[22]，有监督方法将关键词提取看作二元分类的问题，该方法须先提供已标注关键词的训练语料，然后才能对关键词进行判断提取，也就是说要先通过语料训练获得关键词提取模型，然后再基于模型进行关键词提取。半监督方法需要人工参与，非全自动的实现，无需大量的训练数据，只需部分语料数据即可训练关键词提取模型，然后通过训练好的模型可

以对未标注文本进行关键词提取，再人为对抽取结果进行筛选，将正确的标注放入训练语料中再训练，以获得最终关键词。无监督方法无需训练语料，也不需要人工参与，即可实现文本自动关键词提取，因此当训练语料大且无标注信息时，一般使用无监督的关键词提取方法。

本研究中，由于搜集的绝句语料库中并没有关键词的标注数据，而有监督和半监督的方法需要对诗词语料进行关键词标注，若对绝句语料进行人工标注，工作量巨大，过于繁琐且代价过高，因此，本研究选用无监督的方法进行用户写作意图关键词提取。常用的无监督方法主要有 TF-IDF 算法[23]、LDA 主题模型[24]和 TextRank 算法[25]等，TF-IDF 算法单纯以词频判断词语重要性，通常在长文本中效果更好，而对于绝句这种较短的文本不太适用；LDA 主题模型则比较依赖基础语料库，会忽略一些主题性不强却能表征文章要点的重点词，往往需要和其他方法结合起来使用。因此，考虑到以上因素，本研究使用 TextRank 算法对用户写作意图进行关键词提取。

TextRank 算法源于谷歌的 PageRank 算法，是一种基于图的排序算法。PageRank 算法将文本切分成若干组成单元，然后建立图模型，通过投票机制对文本中的词语进行排序，实现关键词抽取，该算法仅需单篇文本自身的信息就能实现关键词提取，其被 Google 搜索用于网页排序，通过计算网页链接的数量和质量来衡量网页重要程度。而 TextRank 算法在进行关键词提取时，首先将文本分成多个文本单元，让文本单元作为节点，而若干文本单元之间的相似度即为节点间的边，从而形成图模型，再通过使用 PageRank 算法对图模型迭代至收敛，最后对所有节点进行排序，输出所求关键词[26]。分以下四步：

（1）将输入文档划分成若干文本单元，添加至图模型，形成节点。

（2）判断节点（文本单元）间的关系，添加到图模型，形成边。

（3）进行算法迭代至收敛。

（4）根据最后收敛时得分，对节点进行排序。

TextRank 算法模型通常用 $G=(V,E)$ 表示，V 表示图中节点的集合，E 表示图中边的集合，E 是 $V\times V$ 的子集，节点 V_i 的得分如下式 2-1 所示：

$$WS(V_i)=(1-d)+d*\sum_{V_j\in In(V_j)}\frac{w_{ij}}{\sum_{V_k\in Out(V_j)}W_{j,k}} \qquad (2\text{-}1)$$

其中，V_i 表示图中节点；d 是阻尼系数 $(0\le d\le 1)$，表示某节点跳转到其他节点概率，通常取值 0.85；W_{ji} 表示节点 V_j 和 V_i 之间边的权重，通常用节点 V_j 和 V_i 的相似度表示；$Out(V_i)$ 表示节点 V_i 连接所有节点集合；$In(V_i)$ 表示指向节点 V_i 的节点集合。使用 TextRank 算法时，通常初始值设定为所有节点得分为 1；另通常收敛阀值设为 0.0001，即在误差率小于 0.0001 时收敛，停止迭代。

在具体运用到关键词提取时，TextRank 算法主要是通过词项之间共现关系进行关键词提取。其步骤如下[25]：

（1）分割文本：将输入文本（即诗歌写作意图）分割成句子，再对句子进行分词及词性标注。

（2）过滤词项：首先将停用词过滤；然后进行词性过滤，保留名词、形容词等特定词性项。

（3）构建图模型：词项作为节点，词项之间的共现关系作为边，若两个词项在长度为 N 的窗口内共现，则判定这两个节点之间存在边，从而构建无向无权图。其中 N 通常取 2 至 10 数值。

（4）代入公式；迭代计算直到收敛。TextRank 算法用于关键词提取时，单词之间并没有权重大小，假设初始值为 1，因此，对公式 2-2 进行修改调整如下：

$$WS(V_i)=(1-d)+d*\sum_{V_j\in In(V_j)}\frac{1}{\left|Out(V_j)\right|}WS(V_j) \qquad (2\text{-}2)$$

（5）输出关键词：对所有节点进行排序，得分最高的单词，即为诗歌主题提取的唯一关键词。

通过以上方式完成写作意图唯一关键词提取后，由于提取的关键词容易出现现代汉语中的白话文词语，而在绝句生成中，是基于绝句语料库诗句中的古代文言文字词，因此现代白话文词语关键词与之不匹配，是不利于绝句生成的，可能会造成生成诗歌语义不畅。为解决该问题，本文提出的绝句生成模型在提取唯一主题关键词后，将先进行文言文转换，然后基于转换后的文言文字词进行主题扩展，生成绝句子主题关键字词，再进行绝句生成。本研究将直接使用百度翻译开放平台的相关接口，进行关键词转换。

百度翻译开放平台是百度翻译面向广大开发者提供开放服务的平台，依托于百度强大的自然语言处理技术和高水准的翻译技术，为广大开发者提供强大、简便、易用的翻译 API 和 SDK 等接口服务。其通用翻译 API 免费支持文言文、中、英、日、韩等 28 种语言互译。只需调用通用翻译 API，输入相应翻译的内容，并设置好源语言和目标语言，就可以得到想要的翻译结果[27]。因此提取的唯一关键词可通过使用通用翻译 API 来实现现代白话文词语转换为文言文词语的翻译服务。百度通用翻译 API 使用方式如下[27]：

（1）使用百度账号登录百度翻译开放平台（http://api.fanyi.baidu.com）；

（2）注册成为开发者，获得通用翻译的 APPID，并进行开发者认证，开通通用翻译 API 服务，开通链接；

（3）通过 HTTP 接口调用翻译 API，传入需要翻译内容，指定源语言和目标语言，即可得到相应翻译结果。其中：

输入：通过 POST 或 GET 方法发送下表 2-1 中字段给通用翻译 API HTTP 地址即可访问服务。

表 2-1 百度翻译输入参数字段（引自百度翻译平台官网）

字段名	类型	必填参数	描述	备注
q	TEXT	Y	请求翻译 query	UTF-8 编码
from	TEXT	Y	翻译源语句	语言列表（可设置为 auto）
to	TEXT	Y	译文语言	语言列表（不可设置为 auto）
appid	TEXT	Y	APP ID	可在管理控制台查看
sait	TEXT	Y	随机数	
sign	TEXT	Y	签名	Appid+q+salt+密匙的 MD5 值

其中，源语言不确定可设置为 auto，但目标语言语种不可设置为 auto，本研究中，因需要进行文言文翻译，根据其相关规则，译文语言字段“to”填“wyw”。“sign”是通过 MD5 算法产生的一段字符串签名，起到调用安全保护作用。

输出：返回结果输出为 json 格式，包含字段如表 2-2：

表 2-2 百度翻译输出参数字段（引自百度翻译平台官网）

字段名	类型	描述	备注
from	TEXT	源语句	返回用户指定的语言，或自动检测的语言（源语言设为 auto 时）
to	TEXT	目标言语	返回用户指定的目标语言
trans_result	MIXEDLIST	翻译结果	返回翻译结果，包括 src 和 dst 字段
src	TEXT	query 原文	
dst	TEXT	译文	
error_code	Int32	错误码	仅当出现错误时显示

例如以下为进行文言文翻译部分代码：

```
{
  "from":"zh",
  "to":"wyw",
  "trans_result":[
    {
      "src":"难过"
      "dst":"悲"
    }
  ]
}
```

只需从中取出 dst 对应的值即为翻译的文言文结果。

2.3.2 关键词扩展

关键词扩展阶段，由于主题关键词有可能在绝句语料库没有覆盖，因此，针对不同情况采取不同的关键词扩展方法：如果是绝句语料库已经出现的关键词，则通过使用基于语言模型的关键词扩展；如果出现绝句语料库中未出现的关键词，则通过引入外部知识的方法进行关键词扩展，接下来将对具体实现进行介绍。

语言模型是根据语言客观事实建立的语言抽象数学模型，是判断语言序列是否是正常的人类语句，是解决自然语言处理领域相关问题的基本模型[28]。语言模型应用广泛，在语音识别、信息检索、舆情分析等方面起到重要作用。统计语言模型是描述自然语言中存在特定规律的数学概率分布模型[29]。其通过给定词来计算词文本中其他词出现的概率，并根据计算得到的词概率，再算出最大概率的词，是解决自然语言上下文相关特性的数学模型。N-gram 语言模型[30]作为一种统计语言模型，将马尔科夫假设引入到词序列概率计算中。假设有句子 $S=\{w_1,w_2,w_3,\cdots,w_n\}$，则该句子出现概率为：

$$P(S)=\prod_{i=1}^{n}p(w_i\left|w_{1:i-1}\right.) \tag{2-3}$$

其中，w_i 为句中第 i 个词，$w_{1:i-1}$ 为句中前 i-1 个词。由式子可知，若要计算句子概率，需计算数据集中所有词序列概率，若数据集大，则复杂度高。N-gram 语言模型则解决了这一问题。在 N-gram 模型中，假设 w_i 的概率仅与前 n-1 个词相关，则有上式简化为：

$$P(S)=\prod_{i=1}^{n}p(w_i\left|w_{i-n+1:i-1}\right.) \tag{2-4}$$

在基本 N-gram 模型中有：

$$P(w_i\left|w_{i-n+1:i-1}\right.)=\frac{count(w_{i-n+1:i})}{count(w_{i-n+1:i-1})} \tag{2-5}$$

该模型中，通常当前词语出现概率与离得近的词语相关性大，远的小。因此 N-gram 语言模型考虑了之前词语的信息，因此之前词语对当前词语约束强，效率更高。但其也存在问题：该模型无法体现词语间的相似度，因此无法评估上下文相似词语出现的概率；当词语间相关距离大于 n 时，无法建模出现稀疏性问题；无法计算语料库没有的词语概率。

随着深度学习技术的发展，基于神经网络的语言模型被提出，其通过使用神经网络方式来计算词序列中词出现的概率，解释自然语言中词句存在特定规律，自动学习代表语法和语义的特征，解决稀疏性问题，并提高泛化能力。Bengio 等人提出前馈神经网络语言模型，该模型通过包含嵌入层、全连接层、输出层三层的全连接神经网络模型来估计给定 n-1 个上文的情况下，计算第 n 个单词出现

概率[28]。其架构图 2-3 所示[31]：

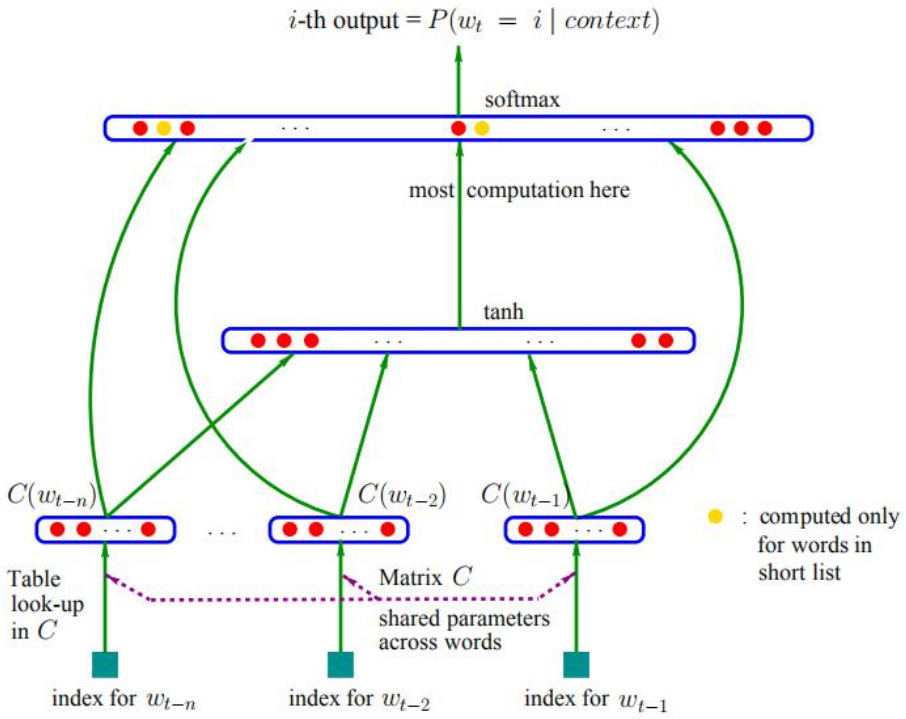

图 2-3：前馈神经网络语言模型架构图（图引自文献[31]）

模型首先将词典中的单词映射到给定维度空间，该映射即图中参数矩阵 C，矩阵行数即为词典中的单词数量，列数即为给定的致密空间的维度，单词在空间的映射就是单词的词向量表达。然后使用激活函数将单词对应上下文映射到词典全部单词对应的条件概率分布空间中。最后，在训练过程中，同时学习词向量的映射关系参数和上下文到单词出现的条件概率参数。通过词向量的映射，前馈神经网络语言模型可以解决稀疏性问题，在实际应用中表现出一定泛化能力，但该模型没有明确对超出观察窗口上下文信息进行处理，仅包含了有限的前文信息。

为了解决定长信息的问题，Mikolov 于 2010 年提出了循环神经网络语言模型（RNNLM）[21]。相比单纯的前馈神经网络，RNNLM 可以捕捉前向序列的所有信息, 通过在训练集上优化交叉熵来训练模型，使得网络能够建立自然语言序列与后续词之间的内在联系。假设一个 V 大小的词表，给定历史序列 $x_{1:t}$，神经网络语言模型可以计算出下一个词 x_{t+1} 的概率分布，RNNLM 通过在隐藏层映射变换，接 Softmax 层实现：

$$P(x_{t+1}=j|x_{1:t})=\frac{\exp(h_t \bullet w^j)}{\sum_{j' \in v}\exp(h_t \bullet w^{j'})} \tag{2-6}$$

其中，w_j 是输出矩阵的第 j 列，RNNLM 以词的向量 $e(x)$ 作为输入，RNN 结构决定隐藏层状态向量 h_t。若 $x_{1:T}$ 是训练预料词序列，则训练目标为最小化序列负对数似然（Nagative Log-likehood,NLL）：

$$NLL = -\sum_{t=1}^{I} \log P(x_t \mid x_{1:t-1}) \tag{2-7}$$

在本研究中，使用 RNNLM 模型进行绝句的主题关键词扩展，通过给定经过文言文关键词转换后的主题关键词 $k_{1:i-1}$ 作为模型输入，可得到下一个关键词 k_i 概率分布，让 $k_i = \arg\max P(k \mid k_{1:i-1})$，则其中概率最大的词即为我们需要扩展的主题关键词，重复执行该过程，直至扩展至 4 个关键词。如图 2-4，将主题提取并经过文言文转换的唯一关键词（雁）输入 RNNLM 关键词扩展模型中，输出其中概率最高的主题词“凉”作为下一个关键词，并将其输入模型中，再次输出概率最高的主题词“故乡”，再将其作为输入放入模型，输出概率最高的主题词“思”，最终绝句每一行得到一个主题关键词，形成绝句完整写作提纲（雁、凉、故乡、思）。

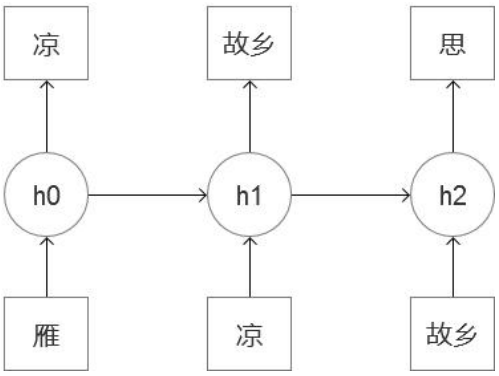

图 2-4 RNNLM 关键词扩展模型图

因本文提出的绝句生成模型中，每一句绝句都需要一个关键词作为子主题，所以在上述基于在 RNNLM 模型的主题扩展中，我们首先需输入收集的绝句语料，将绝句语料中每一句诗进行分词去除停用词，然后通过 TextRank 得到每一句绝句诗的关键词，得到关键词作为模型训练预料，来实现基于现有诗歌预料的关键词扩展。但针对许多人名，地名等词语在经过文言文翻译后，语料库中未覆盖的关键词，无法通过 RNNLM 模型进行关键词扩展，针对这一问题，本文引用王哲的关键词改写和基于外部知识的关键词扩展的方法[32]，该方法借助 Word2vec 及现有语料库，如维基百科语料库、搜狗语料库等，从中提取数据，通过余弦相似度等相似性度量方法，找到与写作意图关键词相近的词语作为关键词。具体扩展有：

关键词改写的方法：（1）基于百科别名的方法：若词语不在诗歌语料库中，但其在百科中存在的别名，则可使用别名代进行关键词扩展。如北京，在古代就有燕京、蓟城、涿郡、幽州等别名。（2）基于同近义词典的方法：若绝句诗预料中未出现，但其同近义词可能出现，因此可将关键词同近义词进行关键词扩展。

外部知识关键词扩展方法：如果使用关键词改写也无法实现关键词扩展，通过引用外部知识源知识图谱、百科等方法，找到与关键词相关词语，再进行关键

词扩展。主要有：（1）外部语料进行 RNNLM 扩展：使用宋词、元曲、散文等其他时期诗歌或者其他文体数据，再用本文中相同的方法进行关键词扩展；（2）百科窗口共现：设定约束条件，使用 TextRank 算法评分选出得分高的词语作为关键词扩展词语；（3）知识图谱扩展方法：不同类型关键词，设计不同模板扩展，如作家名，使用该作家创作的作品名称作为扩展关键词。

2.3.3 绝句生成

绝句生成阶段，主题关键词经过转换与扩展，实现了为每一句绝句诗分配子主题关键词，可看做序列到序列的生成过程，将根据之前提取并转换扩展的子主题关键词和历史生成内容进行绝句生成，本文将使用基于注意力机制的编码器-解码器模型进行绝句诗生成。

编码器-解码器是一种序列到序列的模型，其在编码阶段，将输入序列转化为一个固定向量；解码阶段将编码端输出的固定向量进行解码，转换成输出序列，其输入序列和输出序列长是一样的。该模型广泛应用于图像处理、翻译等领域，是一个框架类的模型，而不是具体的模型，其输入输出可以是多种数据形式，如文本、音频、图像数据等，也可以是多种深度学习模型，如 CNN、RNN、LSTM、GRU 等，其模型框架可抽象表示如图 2-5。

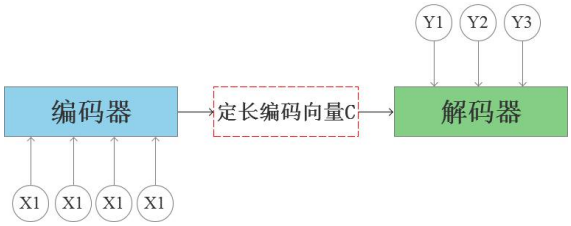

图 2-5 编码器—解码器抽象框架

在自然语言处理领域，编码器-解码器框架可理解为将一个句子变成另一种形式但不改变其含义的处理模型。对于句子对<X,Y>，给定句子输入序列 X，我们可通过模型得到生成目标 Y，其中 X 和 Y 由各自不同单词序列构成：

$$\begin{aligned} X &= (x_1, x_2, \cdots x_m) \\ Y &= (y_1, y_2, \cdots y_n) \end{aligned} \tag{2-8}$$

编码器对 X 进行编码，将输入序列用模型 f 转化为固定长度的中间语义表达式向量 C，其中：

$$C = f(\mathrm{x1}, \mathrm{x2}, \cdots \mathrm{xm}) \tag{2-9}$$

解码器使用中间向量 C 与生成的 $y_1, y_2, \cdots y_{i-1}$ 信息来生成 i 时刻要生成的词 y_i，其中：

$$y_i = g(C, y_1, y_2, \cdots, y_{i-1}) \tag{2-10}$$

每一个y_i都以此方式产生，即可得到目标句子 Y。实际应用中，在翻译领域，如英翻汉，输入为英语句子，则输出即为中文句子；而对于语音识别领域，输入是语音信号，输出就是文字。

编码器-解码器框架应用广泛，但其在生成目标句子的单词时，它使用的语义编码向量 C 始终不变，而向量 C 是由句子 X 中的词语经由编码器编码产生，因此无论生成什么词语，对生成目标句子 Y 中的词语都是一样的影响。由此便导致该框架对于输入的短序列数据影响不大，但对于长序列数据，语义信息始终用唯一中间的语义编码向量 C 表示，就使得细节信息容易丢失。

为了解决上述问题，引入注意力模型，注意力模型由原先固定的中间语义向量 C 变换成了可以根据当前输出词语动态变化的C_i，增加了注意力模型的编码器-解码器框架如图 2-6。

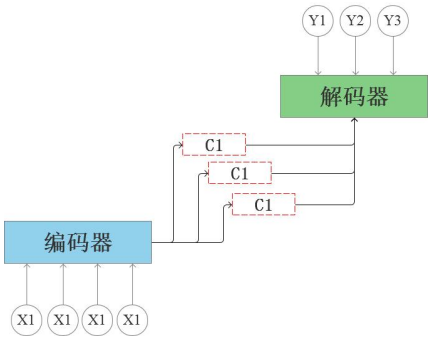

图 2-6 引入注意力模型的编码器-解码器框架

此时生成 i 时刻的词语$y_i = g(C_i, y_1, y_2, \cdots, y_{i-1})$，其中

$$C_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \tag{2-11}$$

T_x为输入序列长度，α_{ij}表示输出第 i 个单词时输入第 j 个单词的注意力权值分配，h_j表示输入第 j 个单词的语义编码。

注意力分配的概率计算常用方法示意图如图 2-7。示意图中采用了 RNN 模型，当计算生成词y_i时，输入词语x_1, x_2, x_3的注意力分配概率分布值，可以通过输出$i-1$时刻的 RNN 隐含层节点状态s_{i-1}去和输入每个单词对应 RNN 隐含层节点状态h_j进行比较比，即用函数$F(h_j, s_{i-1})$来获得目标词语y_i和输入词语的对齐情况，最后通过 softmax 激活函数进行归一化，得到注意力分配概率分布数值。上述为大多数注意力模型常用的注意力分配概率计算方法，其中不同注意力模型其 F 函数的定义会有所不同。

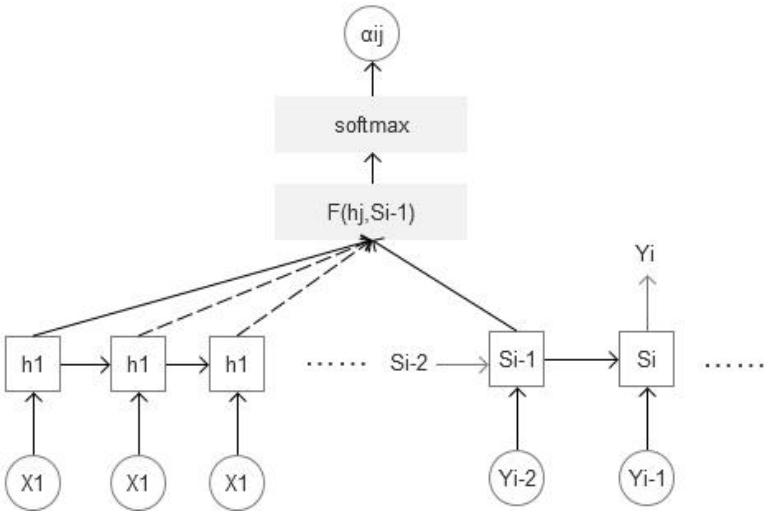

图 2-7 注意力分配概率计算示意图

传统的基于注意力机制的编码器-解码器模型只有一个输入，在绝句生成研究中，除了主题关键词，还有历史生成的内容作为输入，此处参考 PPG 模型的设计，在基于注意力机制的编码器-解码器模型基础上进行了修改，以实现同时将关键词和已经生成诗歌作为诗歌输入，绝句生成模型框架如图 2-8 所示。

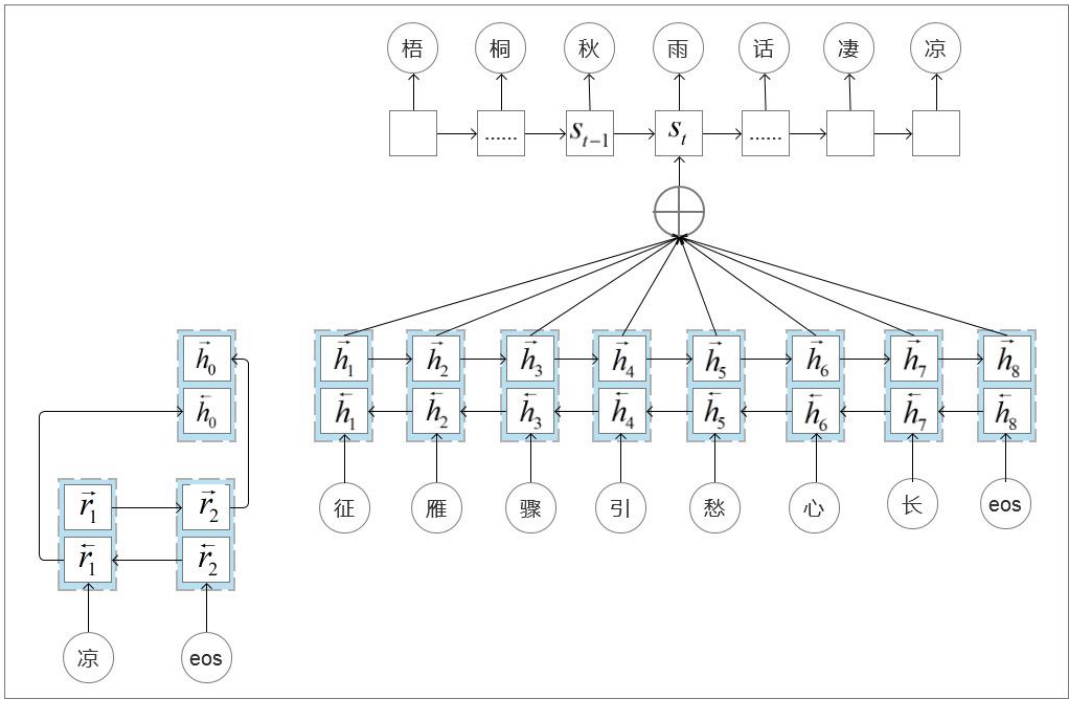

图 2-8 基于注意力机制的编码器-解码器绝句生成模型框架图

假设主题关键词 $\mathrm{k}=a_1,a_2,\cdots,a_{T_k}$ 由 T_k 个字符组成，历史生成的绝句内容 $\mathrm{x}=x_1,x_2,\cdots,x_{T_x}$ 由 T_x 个字符组成。

编码阶段，通过双向 GRU[33]把主题关键词 k 转换为隐状态序列 $(r_1,r_2,\cdots,r_{T_k})$，把历史生成的绝句内容 x 转换为隐状态序列 $(h_1,h_2,\cdots,h_{T_x})$，再将 $(r_1,r_2,\cdots,r_{T_k})$ 最后

一个正向状态向量$\vec{r}_{r_k}$与第一个反向状态向量$\overleftarrow{r}_1$拼接组合起来，合成一个向量r_c。即：

$$r_c = \begin{bmatrix} \vec{r}_{T_k} \\ \overleftarrow{r}_1 \end{bmatrix} \tag{2-12}$$

再设$h_0 = h_{T_x}$，此时新状态向量$\mathrm{h} = (h_1, h_2, \cdots, h_{T_x})$表示绝句主题关键词 k 和历史生成内容 x 的语义信息。当进行绝句生成时，第一行没有历史生成内容，只需要根据第一个关键词进行生成，所以此时$T_x = 0$，$\mathrm{h} = h_0$。

解码阶段，条件概率定义为：

$$p(y_i \,|\, y_1, y_2, \cdots, y_{t-1}, x, k) = \mathrm{g}(y_{t-1}, s_t, c_t) \tag{2-13}$$

$\mathrm{g}(\cdot)$是非线性函数，输出$p(t)$概率，GRU 内部向量s_t按照以下公式进行更新：

$$s_t = f(s_{t-1}, c_t, y_{t-1}) \tag{2-14}$$

$f(\cdot)$是 GRU 的更新公式；y_{t-1}为上一步生成的词；c_t表示背景向量，根据对齐模式按下面公式进行更新：

$$c_t = \sum_{j=0}^{T_h - 1} a_{tj} h_j \tag{2-15}$$

h_j表示输出的第j个隐状态向量，其权值a_{tj}计算公式如下：

$$a_{tj} = \frac{\exp(e_{tj})}{\sum_{k=0}^{T_h - 1} \exp(e_{tk})} \tag{2-16}$$

e_{tj}为当h_j在第t步时候的得分，有：

$$e_{tj} = v_a^T \tanh(W_{a^{S_{t-1}}} + U_a h_j) \tag{2-17}$$

最后，模型的参数使用最大化对数似然的方式进行更新：

$$\arg\max \sum_{n=1}^{N} \log P(y_n \,|\, \mathrm{x}_n, \mathrm{k}_n) \tag{2-18}$$

2.4 评价体系

因绝句的特殊文体结构及其格律性等要求，通常很难对绝句生成模型进行简单的量化评估，本文针对模型及绝句的相关特点，设计了自动评估、人工评估及图灵测试来验证模型的有效性。下面对本文的相关评价方法进行介绍。

2.4.1 BLEU 自动评估

BLEU(bilingual evaluation understudy)是一种常用于机器翻译的评估算法，用于定义机器翻译译文和参考译文之间的相似度，当机器翻译译文越接近人工参考翻译结果，则说明翻译质量就越高。BLEU 采用 N-gram 的匹配规则来比较机器翻译译文和参考译文之间 n 组词相似占比。

若人工参考译文用 reference 表示机器翻译译文用 candidate 表示，最早的 BLEU 算法计算方式可以表示如下：

$$\text{BLEU}=\frac{\text{出现在参考译文中的机器译文的单词个数}}{\text{机器译文中单词总数}} \tag{2-19}$$

例如：

原文：我今天吃了一个苹果

机器译文：I eat a apple today

参考译文：I ate an apple today

机器译文中有 5 个单词，有 3 个单词和参考译文一样，则其 BLEU 值为 3/5。但该计算方式明显是有问题的，如当机器译文为“eat eat eat eat eat”时，其都在参考译文中出现过，此时 BLEU=5/5，所以这种统计分子的方式是不合理的。针对该问题，对分子取值进行改进，具体做法如下：

$$Count_{w_i}^{clip}=min(Count_{w_i},Ref\_Count_{w_i}) \tag{2-20}$$

其中，$Count_{w_i}$ 表示单词在 w_i 在机器翻译译文中出现的次数，$Ref\_Count_{w_i}$ 表示单词 w_i 在人工参考译文中出现的次数，通常参考译文会有多个，所以有：

$$Count^{clip}=max(count_{w_i,j}^{clip}),j=1,23\cdots \tag{2-21}$$

其中，j 表示第 j 个参考译文，在此限制下，当分子出现五个“eat”时，则只计算一个，所以此时 BLEU=1/5。

在进行单词翻译时，一个单词进行比较可看作是 1-gram，该方法可描述机器的逐字翻译的能力，但无法判断翻译的流畅性，因此引入了 n-gram，其中通常 n 不大于 4。引入 n−gram 后其精度表示如下：

$$p_n=\frac{\sum_{c\in candidates}\sum_{n\text{-}gram\in c}Count_{clip}(n\text{-}gram)}{\sum_{c'\in candidates}\sum_{n\text{-}gram'\in c'}Count_{clip}(n\text{-}gram')} \tag{2-22}$$

在评价机器翻译质量时，通常会使用多条机器翻译译文来评价，因此上式中引入候选集合 candinates。p_n 表示中的 n 表示 n-gram，即 n=1，则表示 1-gram；

第一个Σ表示各个机器翻译译文的总和，第二个Σ表示一条机器翻译译文中所有n-gram的总和；$Count_{clip}(n\text{-}gram)$表示其中一个n-gram词的截断计数，$Count_{clip}(n\text{-}gram')$表示$n\text{-}gram'$在机器翻译译文中的计数，简单来说，分母即为机器翻译译文中n-gram个数，分子就是出现在人工参考译文中的机器翻译译文中的n-gram个数。例如，若用2-gram进行评估，如图2-9，上面为机器翻译译文，下面为人工参考译文，翻译译文分4个2-gram的词组，其中有1个和参考译文一样，则它的2-gram匹配度为1/4。

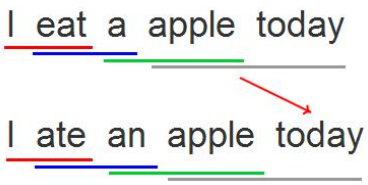

图2-9 2-gram评估示意图

依次类推，即可以实现一个程序来遍历计算N-gram的匹配度。对于翻译中出现译文很短的句子时，通常会有较高的BLEU值，因此引入对句子长度的乘法因子，其表达式如下：

$$\mathrm{BP}=\begin{cases}1 & if\ c>r\\ e^{(1-r/c)} & if\ c\le r\end{cases} \tag{2-23}$$

其中，c表示机器翻译译文长度，r表示人工参考译文长度，将以上整合，得到BLEU最终表达式如下：

$$BLEU = BP\exp(\sum_n^N w_n \log p_n) \tag{2-24}$$

因本文提出的绝句生成模型是基于前一句历史生成诗歌和子主题进行生成，其生成过程一定程度上跟机器翻译类似，由于绝句生成不太可能产生与训练诗歌预料完全相同的诗，这不是目标函数的工作原理，所以BLEU只能大致的校准诗歌的生成能力，反映生成诗歌跟训练预料的相似性，但其结果同样具有一定的指导意义，可以一定程度上反映生成诗句的流畅性。因此对于绝句生成模型的评价，借鉴BLEU算法进行自动评估[34]。在该方法中，BLEU取值范围在0和1之间，当译文和参考译文一模一样时，其取值才能为1，因此BLEU值越大说明翻译模型越好，当生成的诗句贴近训练语料的参考下句时，则BLEU值越大，则认为诗歌生成效果更好。

因使用BLEU进行评价，需要给测试集中的诗句设置参考诗歌，本文借鉴He等人的方法[12]。即若两首绝句具有类似的主题，则它们可以作为作为彼此参考诗歌。因绝句生成为五言和七言，其长度相等，且绝句诗基本元素为字或二字

词语，所以 BP 值始终为 BLEU 指标统计到 bi-gram（即 2-gram）。

2.4.2 人工评估

对于诗歌生成质量的评估，通常没有较为有效的自动评估方法，现有的对于诗歌生成质量的评估，主要以人工评估的方式进行，通常采用基于流畅性、连贯性、格律性、意义四个评价标准进行人工评价[11][12][13]，其主要考虑生成诗歌的流畅性、连贯性、格律性和意义四个标准，即：格律是否合格、诗句是否流畅、诗句间是否连贯相关、诗句是否有内容有意义。结合绝句的文学方面相关研究，发现只从这四个标准对绝句进行评价是片面的，以上四个标准可以说是生成的句子可称作诗歌的最基本要求，而一首合格的诗歌，除了语义连贯平仄押韵等基本要求外，诗歌的意境/情感更为重要，情境交融才是评价诗歌最恰当的标准[35]。另外合格的诗歌通常可以很好的切合主题，因此评价创作的诗歌好坏，切合主题，点题同样必不可少。

因此本文在现有诗歌人工评估标准基础上，加入意境/情感和切题两个评价指标，可实现更全面合理的人工评估。改进后的人工评估相关标准如表 2-3 所示。

表 2-3 人工评估标准细则

评价标准	说明
流畅性	生成诗歌的诗句是否自然流畅？
连贯性	生成诗歌的诗句之间是否前后关联？
格律性	生成的诗歌诗句是否满足古诗的格律要求？
意义	生成的诗歌是否真实表达内容及具体实际意义？
意境/情感	生成的诗歌是否具有情感和诗歌意境？
切题	生成的诗歌是否切合写作主题？

在具体的人工评估过程中，要求评估人员具有诗词专业背景，然后根据上表，分别从“流畅性”、“连贯性”、“格律性”、“意义”、“意境/情感”、“切题”六个评价标准进行打分，然后取评估分数均值作为评价诗歌生成模型的最终分数，以此来评价生成绝句的好坏。

2.4.3 图灵测试

图灵测试的概念由艾伦·麦席森·图灵在 1950 年所写的《计算机与智能》一文中提出[36]，图灵将该测试作为判别计算机是不是拥有正常人类智能的标准。图灵测试通过把人和计算机隔离，然后对参与测试的人提问，让其判别哪一个是计算机哪一个是人类，如果在该问答判断过程中，错误率超过了 30%，则认为计算机通过了测试，拥有了人类的智能，而这一测试方式就称为图灵测试[37]。因此在本文的绝句生成研究中，若通过图灵测试，则可以证明本文提出的绝句生成模

型达到了普通人类的创作水平，该实验对于评估绝句生成好坏具有可行性及参考价值。

本文实验设计如下：首先通过抽取任意数量人类创作的绝句进行主题概括，得到相关主题词作为输入，再由 KTEQG 模型生成同一主题的绝句，由此得到两首同一主题要求的绝句，将计算机生成的诗歌和人类创作诗歌组成测试卡片，随机标记为绝句一和绝句二，评测人员依次对卡片内绝句进行判断。如图 2-10 为测试示列卡片，参与测试者需从以下三个选项中进行判断：A、绝句一为人类创作；B、绝句二由人类创作；C、两者无法判断。对测试结果进行统计时，若把计算机生成的绝句诗判断为人类创作的，则记为错误，否则记为正确。

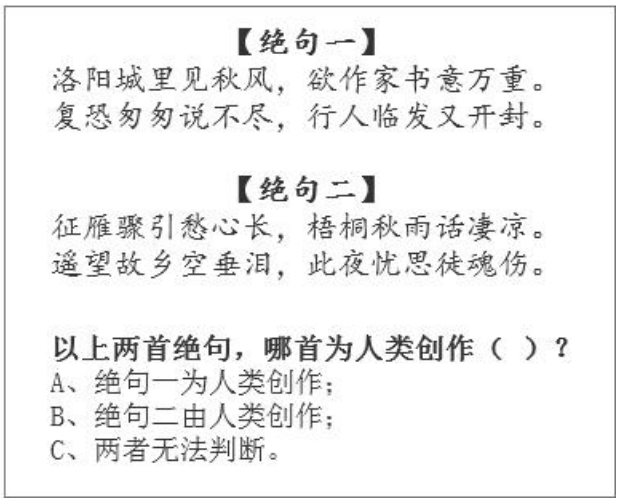

图 2-10 基于关键词转换扩展的绝句生成模型图灵测试示列卡片

通过对测试卡片进行统计，得出最终判断的正确率，错误率和无法判断率，以进行图灵测试结果分析。

2.5 本章小结

本章首先对绝句生成这一深度学习问题，进行了一般性描述；然后对基于规则和模板的生成方法、基于统计机器学习的方法和基于深度学习的方法三个阶段的绝句生成相关工作进行了研究讨论；接着针对现有绝句生成模型中存在的主题漂移、生成诗句语义不连贯等问题，提出了基于关键词转换扩展的绝句生成模型（Keyword Transformation and Expansion Quatrain Generation Model，KTEQG），然后依次从关键词转换、关键词扩展、绝句生成三个方面展开详细介绍其具体实现。最后针对本研究中绝句生成的评估体系进行了介绍，本研究将从自动评估、人工评估、图灵测试三方面对模型进行评估。

3 绝句生成模型算法的实施

3.1 文本的表示

自然语言处理的问题要转化为机器学习问题，首先要考虑的就是如何把语言符号数学化，即词语如何在计算机中表示[38]。在诗歌自动生成的研究任务中，首先需要做的就是将诗句转化为计算机能够“读懂”处理的形式，常见的文本表示方法有 WordNet、独热编码、共现矩阵、TF-IDF、Word2vec[39][40]等方法。

WordNet[41]是文本表示的经典方法之一，它主要依赖外部的词汇知识库对给定单词的定义、同义词、祖先、派生词等信息进行编码，推断给定单词的各种信息，因此，WordNet 其实就是一个词汇数据库，用于对数据库中单词之间的词性标签关系进行编码。WordNet 由美国 Princeton 大学的心理系创建，其通过单词之间的同义性来判断单词之间的关系。适用于英语的 WordNet 建立最为丰富，现阶段已经拥有 150000 个单词 100000 个同义词组，当然，WordNet 不限于语言，许多公司或组织也致力于中文 WordNet 的建设。

WordNet 通常用同义词集表示一组同义词，每个同义词集都会有一个定义，用于解释同义词集所表示的内容。其中的单词表示是通过分层建模的，它在给定的同义词集之间关联形成复杂的图，WordNet 中有 is-a 和 is-made-of 两种关联方式。Is-a 关系：对于给定的同义词集，存在上位词和下位词两类关系，其中上位词是指所有的同义词集中更高一层含义的同义词。例如，animal 是同义词 dog 的上位词。下位词指的是比相应的同位词组更详细具体的同义词。例如，husky 是同义词 dog 的下位词。is-made-of 关系：一个同义词集的整体词可以表示这个同义词集的全部实体的同义词组。例如，poplar 的整体词是 tree。如果部分词是 is-made-of 类别的关系，则它是整体词的反义词，其中部分词是组成相应同义词集的一部分。

WordNet 中的同义词集的不同关联关系如图 3-1 所示。虽然 WordNet 在某些问题上表现出色，在很多自然语言处理任务中都可以用它学习单词的含义，但它同样有许多不足之处：首先其缺少细微差别，从理论角度上看，其对于两个实体间微妙差异的定义进行建模是主观的，例如单词 like 和 love 具有相似的含义，但是其两者实际使用存在很多差别，这即是一种细微差别。其次 WordNet 本身是主观的，其是由相对较小的社区设计，其是否合适，取决于你要解决的问题。然后 WordNet 再维护和跨语言开发中，需要大量人力及迁移成本。

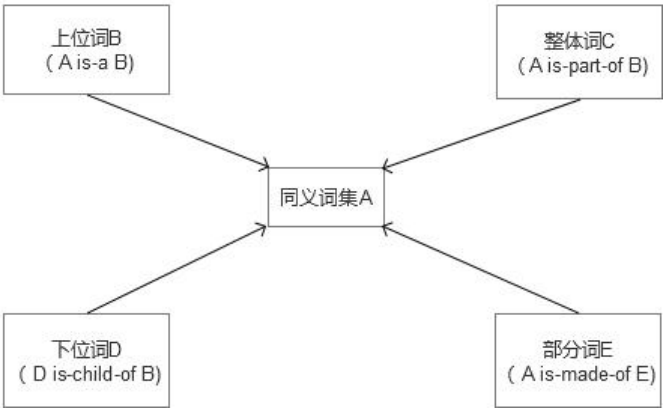

图 3-1 一个同义词集的不同关联

相比 WordNet，独热编码（one-hot）是一个更简单的文本表示方法。假设有大小为 V 的词汇表，其第 i 个词 w_i，可以用长度为 V 的向量[0,0,0，…，0,1,0，…，0,0]表示，这个向量中，第 i 个元素为 1，其他元素全部为 0，这便是独热编码的表示方式。该方法虽然简单，但是也存在许多问题：首先独热编码表示方法没有对单词之间的相似性进行标记，忽略了单词的上下文。然后当词汇表单词量变大时，比如当词汇量超过 20000 个单词，则 20000 个单词的表示矩阵将需要非常稀疏的 20000×20000 的矩阵。后续出现的共现矩阵和独热编码不同，它会考虑单词上下文信息进行编码，但是要维持矩阵大小，矩阵大小会出现随着词汇量大小多项式增长的问题。

TF-IDF 根据单词在语料库中出现的频率进行表示，是一种可以表示文档里特定词语的重要性的表示方法，文档中的词语频率高，则表明该词语在文档中更为重要。对于像 is 这类没有表示很多信息的单词，TF-IDF 通过把这些单词置 0 来处理[42]。在该方法中，TF 为词频率，IDF 为逆文档频率，其计算方式如下：

$$
\begin{aligned}
&TF(w_i) = w_i \text{ 出现的次数/总单词数} \\
&IDF(w_i) = \log(\text{文件总数/带有}w_i\text{的文档数}) \\
&TF-IDF(w_i) = TF(w_i) \times IDF(w_i)
\end{aligned}
\tag{3-1}
$$

TF-IDF 方法具有实现简单易理解等优点，但是其缺点也很明显：该方法不能反映词的位置信息，十分依赖语料库的质量，且需要选取语料库需要与所处理文本相符。而对于 IDF 来说，其自身试图抑制噪声的加权，文本中频率小的词容易被获取，由此造成 TF-IDF 算法的准确度不高。

Word2vec[39][40]是一种分布式单词表示学习技术，Word2vec 通过查看单词上下文通过以数字方式学习表示给定单词的含义。其中“上下文”是指需要表示单词前后固定数量的单词。假设语料库有 N 个单词，通过 $w_0, w_1, \cdots, w_i$ 和 w_N 表示一系列单词，其中，语料库里的第 i 个单词可以表示为 w_i。若给定任意的单词，可

以正确预测上下文单词，则有：

$$p(w_{i-m},\cdots,w_{i-1},w_{i+1},\cdots,w_{i+m}\,|w_i)=\prod_{j\neq i\Lambda j=i-m}^{i+m}p(w_j\,|w_i) \tag{3-2}$$

其中，要实现等式，需假设给定单词 w_i 的上下文单词是彼此独立的。

Word2vec 主要有两种模型结构，分别为 CBOW 和 Skip-Gram 模型，CBOW 是通过词语的上下文来计算该词语出现的条件概率，而 Skip-Gram 是通过词语来计算该词语上下文中的词语出现的条件概率。两种模型示意图如图 3-2。

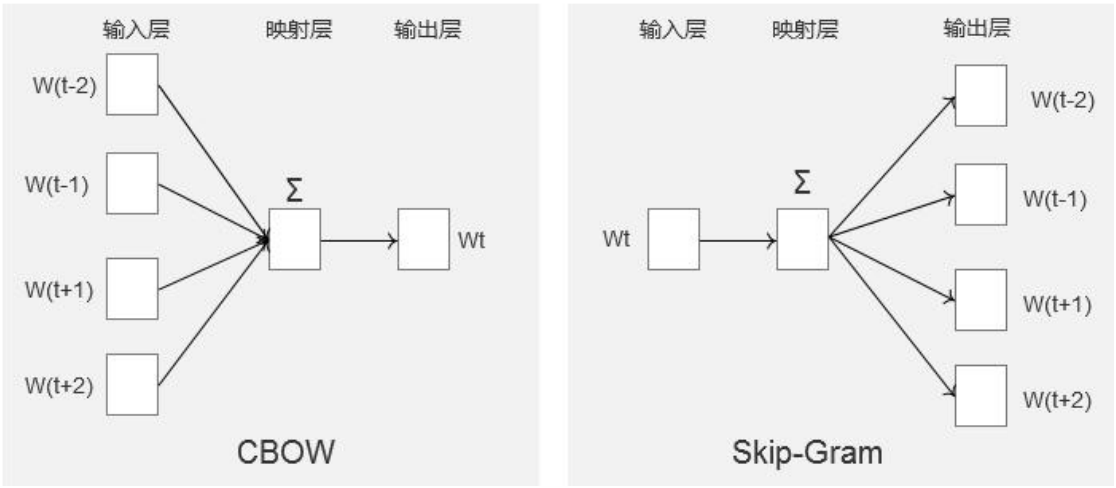

图 3-2 CBOW、Skip-Gram 模型示意图

Word2vec 通过神经网络根据给定的单词来预测上下文单词，使得神经网络被迫学习良好的词嵌入，该方法与先前描述的方法相比有许多优点：

（1）相比 WordNet 方法，Word2vec 对人类语言不具有主观性。

（2）与独热编码和共现矩阵不同，Word2vec 不受词汇量大小影响。

（3）Word2vec 作为一种分布式表示，它的单词表示由向量中所有元素的激活状态决定，相比独热编码向量表示取决于单个元素激活状态而言，Word2vec 具有更强的表达能力。本文在关键词扩展阶段，便借助了 Word2vec 方法。

3.2 神经网络的选择

随着深度学习技术的发展，越来越多的神经网络结构开始应用于自然语言处理领域，如卷积神经网络（Convolutional Neural Networks，CNN）[43]、循环神经网络（Recurrent Neural Network，RNN）[44]、长短期记忆网络（Long Short-Term Memory，LSTM）[45]等，不同的神经网络结构在不同的应用中表现通常表现出不一样的特性。针对绝句生成这一具体实验，我们在神经网络结构的选择问题上进行了细致研究，对自然语言处理领域常用的神经网络进行分析，并从中选择最适合本实验的神经网络。

CNN 是一种具有深度结构且包含卷积计算的前馈神经网络，通常有一个及以上的卷积层、池化层或全连接层，卷积层通过卷积操作将输入提取特征并传入下一层，从而让神经网络可以通过更少的参数实现更深的深度，Kim Y 将 CNN 用于文本分类任务中，基于 Word2vec 搭建 CNN 模型，通过实验比较，证明卷积神经网络在自然语言处理领域同样表现出色[46]，CNN 更多的应用于图像处理等领域。

RNN 是一种以序列数据为输入，在序列的演进方向进行递归且所有循环单元按链式连接的递归神经网络（Recursive Neural Network），其网络输出不仅仅依靠当前的输入，而且还有前一步的神经元状态，其具有记忆性、参数共享并且图灵完备，对序列的非线性特征进行学习时具有一定优势。

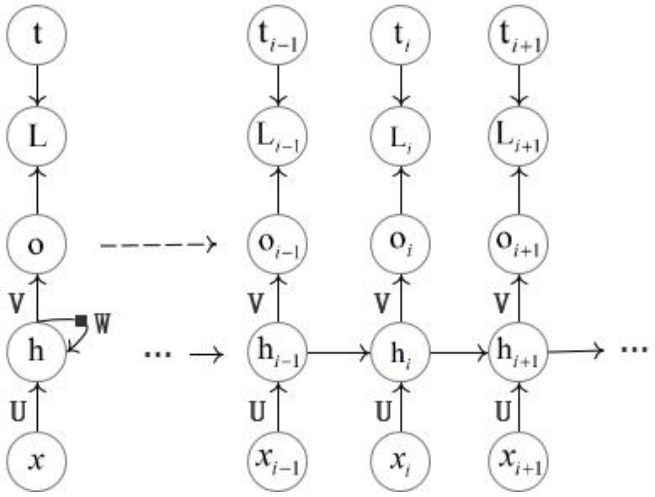

图 3-3 循环神经网络

如图 3-3，循环神经网络由隐藏状态 h 和可选择使用输出 o 构成，V 为隐藏状态 h 和输出 o 之间的权重矩阵，W 是隐藏状态到隐藏状态的权重，U 是输入和隐藏状态之间的权重矩阵，l_i 为每一步的损失，t_i 为每一步的训练目标。循环神经网络的隐藏状态 h_i 随时间步 i 更新，如下式 3-3：

$$h_t = f(h_{t-1}, x_i) \tag{3-3}$$

由式子可知，当前隐藏状态 h_i 根据上一时间隐藏状态 h_{t-1} 和当前时间输入 x_i 决定，其中 f 是非线性激活函数。在实际的使用过程中，f 可以根据实际需要，选择 sigmoid 或 tanh 等函数。

RNN 可根据序列中当前时刻的词语信息预测下一个时刻词语，因此可以学习到全部序列上的联合概率分布。通常在时间步 i 时刻，模型输出的即为条件概率 $p(x_i \mid x_{i-1}, \cdots, x_1)$，也就是说，模型输出是在给定前 $i-1$ 个词语 $\mid x_{i-1}, \cdots, x_1$ 时，生成词 x_i 的条件概率。如果训练语料数据中词语遵循多项分布，输出可由 softmax 函数表示，对所有可能词 $k = 1, \cdots, K$ 有：

$$p(x_{ik}=1|x_{i-1},\cdots,x_1)=\exp(v_k h_t)/\sum_{k'=1}^{K}\exp(v_{k'}h_i) \quad (3\text{-}4)$$

上式中，v_k 为隐藏状态 h_i 到 o_i 权重矩阵 V 的第 k 行，首先计算所有时刻条件概率，然后利用贝叶斯原理，便可计算出整个序列的联合概率分布：

$$p(x)=\prod_{i=1}^{T_x}p(x_i|x_{i-1},\cdots,x_1) \quad (3\text{-}5)$$

循环神经网络训练过程中，其梯度通常在进行多次传播后，会出现梯度消失问题，因此将阻止模型学习长期依赖关系。因此针对这一问题，后续出现了长短期记忆网络。

LSTM 基于 RNN，加入了自循环，自循环权重可以根据上下文确定，LSTM 通过控制门控制该权重，使得单元状态可以动态改变[45]。

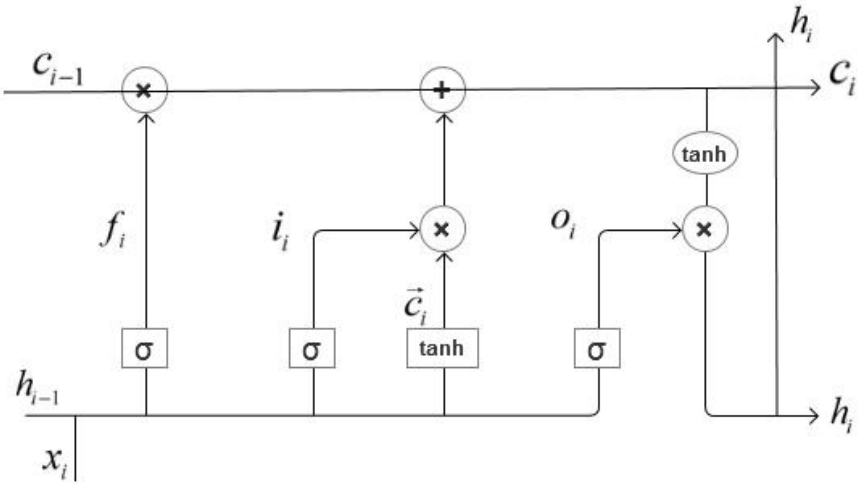

图 3-4 长短期记忆网络

如图 3-4，LSTM 的关键是单元状态 C_i，它像一条传送带，沿着模型整个信息链进行传递，传递过程中，通过简单线性处理，可以实现 LSTM 添加或删除信息到单元状态中，而这个添加或者删除操作，通过控制门来实现。门结构由 sigmoid 激活单元和逐点乘法计算组成，sigmoid 激活单元只输出 0 或 1，值为 0 时门关闭，信息无法通过，值为 1 时门打开，所有信息通过。LSTM 的门机制由遗忘门、输入门、输出门组成：

遗忘门：遗忘门用来控制单元状态 C_i 中哪些信息保留，哪些信息丢弃。遗忘门 f_i 为 sigmoid 激活函数，由上一时刻隐藏状态 h_{i-1} 和当前时刻输入 x_i 作为输入，其输出与上一时刻单元状态 C_{i-1} 维度对应，输出值在 0 和 1 之间，1 表示保留上一时刻信息，0 表示将上一时刻单元状态丢弃。遗忘门可表示为：

$$f_i=\sigma(W_f[h_{i-1},x_i]+b_f) \quad (3\text{-}6)$$

其中 $[h_{i-1},x_i]$ 为上一时刻隐藏状态 h_{i-1} 和该时刻输入 x_i 的连接结果。W_f 和 b_f

是$[h_{i-1},x_i]$的权重和偏置项。

输入门：输入门用来控制当前时刻生成的新信息哪些传入单元状态C_i中。首先输入门由 sigmoid 函数i_i决定要保留哪些信息，然后将上一时刻隐藏状态h_{i-1}和该时刻输入x_i作为输入，通过 tanh 激活函数生成新的候选状态向量$\tilde{C}_i$。最后通过逐点乘积生成最终的保留信息。输入门i_i和候选状态向量$\tilde{C}_i$表示如下：

$$i_i = \sigma(W_i[h_{i-1},x_i]+b_i) \tag{3-7}$$

$$\tilde{C}_i=\tanh(\mathrm{W}_C[h_{i-1},x_i]+b_C) \tag{3-8}$$

其中，W_i和W_C为权重矩阵，b_i和b_C为偏置项。

由上述遗忘门和输入门，我们得到了需要丢弃的信息和需要保存的信息，我们通过上一时刻单元状态C_{i-1}和遗忘门f_i逐点相乘得到C_{i-1}丢弃信息，通过候选状态向量$\tilde{C}_i$和输入门i_i逐点相乘得到需要保存的信息，对于诗歌生成任务来说，上述操作其实就是将上一时刻上下文信息添加到当前时刻单元状态C_i中，可组合表示为：

$$C_i = f_i \cdot C_{i-1} + i_i \cdot \tilde{C}_i \tag{3-9}$$

输出门：输出门用来控制最终哪些信息输出。信息输出阶段，先由输出门 sigmoid 函数o_i决定输出哪些信息，单元状态C_i由 tanh 激活函数确定候选输出，然后将候选输出与输出门o_i逐点相乘，得到该时刻隐藏状态h_i。输出门o_i可表示为：

$$o_i = \sigma(W_o[h_{i-1},x_i]+b_o) \tag{3-10}$$

该思科隐藏状态h_i可表示为：

$$h_i=\mathrm{o}_i \cdot \tanh(C_i) \tag{3-11}$$

其中，W_o为权重矩阵，b_o为偏置项。

LSTM 是 RNN 的一个优秀的变种，其继承了大部分 RNN 模型的特性，通过门机制，可以很好地控制信息的保存与丢弃，解决了梯度反传过程由于逐步缩减而产生的梯度消失问题，使其可以存储更长的记忆。但 LSTM 存在计算费时问题，每一个 LSTM 的单元状态里面都意味着有 4 个全连接层, 如果 LSTM 的时间跨度很大，并且网络又很深，这个计算量会很大，且耗时。

本研究在诗歌生成阶段，选择使用的门控循环单元（Gated Recurrent Unit, GRU）[36]可以视为 LSTM 的简化版，是 LSTM 的一种变体。

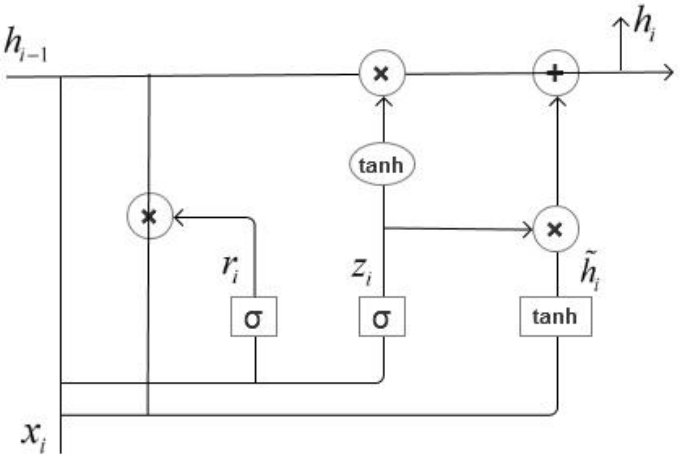

图 3-5 门控循环单元

如图 3-5，门控循环单元把输入门和遗忘门进行合并，组成更新门，同时将单元状态和隐藏状态进行合并，相比 LSTM，门控循环单元可以通过单个更新门同时控制遗忘信息和更新状态，其可用以下公式表示：

$$h_i = (1 - z_i) \cdot h_{i-1} + z_i \cdot \tilde{h}_i \tag{3-12}$$

其中，$\cdot$ 表示逐点乘积运算，z_i 为更新门，可表示为：

$$z_i = \sigma(W_z[h_{i-1}, x_i]) \tag{3-13}$$

上式中，W_z 为 z_i 和 $\tilde{h}_i$ 对应权重矩阵。更新门可同时控制上一时刻隐藏状态 h_{i-1} 中需要被丢弃的信息和新的候选隐藏状态 $\tilde{h}_i$ 中需要保存的信息。新的候选隐藏状态 $\tilde{h}_i$ 表示为：

$$\tilde{h}_i = \tanh(W_a[r_i \cdot h_{i-1}, x_i]) \tag{3-14}$$

其中，$[r_i \cdot h_{i-1}, x_i]$ 为上一时刻隐藏状态 h_{i-1} 和该时刻输入 x_i 连接结构，W_a 为 z_i 和 $\tilde{h}_i$ 对应权重矩阵，r_i 为复位门，复位门表示为：

$$r_i = \sigma(W_r[h_{i-1}, x_i]) \tag{3-15}$$

其中，W_r 表示与复位门 r_i 相对应的权重矩阵。

门控循环单元可解决标准 RNN 中的梯度消失或者爆炸问题并同时保留序列的长期信息，保持了 LSTM 的效果同时又使结构更加简单，是目前自然语言处理领域流行的网络结构，因此本研究在诗歌生成阶段便选择了门控循环单元进行解码。

3.3 优化算法的选择

深度学习的反向传播过程通常十分复杂，针对不同的实验，常常需要反复进

行调参，调参过程常常让人头疼，这一问题的出现，便衍生出许多相关优化算法，如随机梯度下降、Adam 等。而不同的优化算法在实际应用中，与表现出不一样的特性。针对诗歌生成研究的具体需求，本节对常见的优化算法进行分析，并从中选择最适合绝句生成模型的优化算法。

3.3.1 随机梯度下降

随机梯度下降（Stochastic Gradient Descent，SGD）[47]求解过程中，通过从样本中抽取任意样本，进行损失和梯度计算，然后更新参数，计算过程中，只需求解损失函数的一阶导数，并且不需要每一次都遍历所有的数据，因此，其计算的迭代速度很快，可以在规模数据集上应用，但 SGD 的求解是局部最优值，也就是说，计算得到的结果不一定是全局最优步长选择，要么步长过小使得函数收敛速度慢，要么步长过大导致找不到最优解。与之对应的批量梯度下降（Batch Gradient Descent，BGD）在每一次迭代时都使用所有样本来进行损失计算和梯度的更新，其一次迭代是对所有样本进行计算，利用矩阵进行操作实现了并行，相对 SGD 更稳定，BGD 由全部数据集来确定方向，可以很好的代表样本总体，使得训练可以更准确地朝向极值所在的方向，当目标函数为凸函数时，BGD 一定能够得到全局最优。但因为每次都需要计算所有数据，当样本量大时，其训练速度会变得非常慢。小批量梯度下降（Mini-Batch Gradient Descent，MBGD）可以说是介于 SGD 和 BGD 之间的算法，其通过随机抽取 m 个样本，用 m 个样本梯度均值近似估计整个样本梯度[48]，在深度神经网络的训练中，常常会使用到 MBGD，它具有更快的训练速度，但是其训练常常出现结果并不是最终最优解，而是在最优解附近徘徊；另外在 MBGD 算法中，学习率的选择至关重要，如果学习率过大则虽然训练速度快，但可能无法达到最优解，如果学习率过小，虽然训练结果准确，但训练速度极慢，影响效率，而找到一个最佳的学习率却并非易事，因此在诗歌生成研究中，模型的训练没有考虑该方法。

3.3.2 动量

在神经网络训练中，随机梯度下降算法虽受欢迎，但其存在学习过慢，学习率难确定，易错过最优解等问题，而动量（Momentum）[49]优化算法在 SGD 的基础上, 加入了历史梯度更新信息，它可以实现良好的收敛速度，同时解决学习过程摆动幅度过大的问题。该方法基于梯度的移动指数加权平均，它不仅基于当前点的梯度，还考虑历史上梯度，两者加权求和作为当前点的新梯度，从而实现让梯度摆动幅度变小。假设神经网络在迭代步骤第 t 步中，动量优化算法可表示为：

$$\boldsymbol{v}_{dw} = \beta \boldsymbol{v}_{dw} + (1-\beta) dW \tag{3-16}$$

$$\boldsymbol{v}_{db} = \beta \boldsymbol{v}_{db} + (1-\beta) db \tag{3-17}$$

$$W = W - \alpha \boldsymbol{v}_{dw} \tag{3-18}$$

$$b = b - \alpha \boldsymbol{v}_{db} \tag{3-19}$$

上面公式中，v_{dw}和v_{db}表示损失函数在前$t-1$轮迭代中累积的梯度动量，β一般设为0.9，表示梯度累积指数。dW和db是损失函数反向传播过程中所得的梯度，式3-8为权重向量更新公式，式3-9为偏置向量的更新公式，α为学习率。动量优化算法可以解决Mini-Batch SGD算法的损失函数更新幅度摆动过大问题，同时具有较好的收敛速度。

3.3.3 RMSProp

由上一小节我们了解到动量可以加快梯度下降，还有一种RMSprop（Root Mean Square Prop）[50]的算法，也可以加速梯度下降，在动量优化算法中，一定程度上解决了损失函数更新幅度摆动过大问题，为进一步优化收敛速度和摆动幅度过大问题，Geoffrey E. Hinton提出了RMSProp算法。该算法参数的平方和采用指数加权平均。假设神经网络在迭代步骤第t步中，RMSProp算法可表示如下：

$$\boldsymbol{S}_{dw} = \beta \boldsymbol{S}_{dw} + (1-\beta) dW^2 \tag{3-20}$$

$$\boldsymbol{S}_{db} = \beta \boldsymbol{S}_{db} + (1-\beta) db^2 \tag{3-21}$$

$$W = W - \alpha \frac{dW}{\sqrt{\boldsymbol{S}_{dw}} + \varepsilon} \tag{3-22}$$

$$b = b - \alpha \frac{db}{\sqrt{\boldsymbol{S}_{db}} + \varepsilon} \tag{3-23}$$

以上公式中，s_{dw}和s_{db}表示损失函数在前$t-1$轮迭代中累积的梯度动量，β为梯度累积指数，和动量优化算法不一样的是，RMSProp算法的梯度为微分平方加权平均数。这有利于矫正摆动幅度，从而保证每个维度的摆动都保持较小幅度，并且实现网络更快的收敛。同时为避免分母为零，设置了ε参数进行平滑，一般取值为10^{-8}。

3.3.4 Adam

上述中的动量算法通过类似物理中的动量来累积梯度，RMSProp优化算法则可实现摆动幅度小且收敛速度快，Adam（Adaptive Moment Estimation）优化算法[51]可以说是动量优化算法和RMSProp优化算法的结合，该方法所使用的参数基本和动量及RMSProp优化算法一致，但Adam算法在训练前，需要初始化梯度和平方累积量：

$$v_{dw}=0,v_{db}=0;s_{dw}=0,s_{db}=0 \tag{3-24}$$

假设神经网络在迭代步骤第t步中，可通过下式计算出动量算法和 RMSProp 算法的参数更新：

$$\boldsymbol{v}_{dw}=\beta_1\boldsymbol{v}_{dw}+(1-\beta_1)dW \tag{3-25}$$

$$\boldsymbol{v}_{db}=\beta_1\boldsymbol{v}_{db}+(1-\beta_1)db \tag{3-26}$$

$$\boldsymbol{s}_{dw}=\beta_2\boldsymbol{s}_{dw}+(1-\beta_2)dW^2 \tag{3-27}$$

$$\boldsymbol{s}_{db}=\beta_2\boldsymbol{s}_{db}+(1-\beta_2)db^2 \tag{3-28}$$

在迭代初期，移动指数平均将会导致与迭代初期的值出现较大偏差，所以需对上面式子求得的若干值做偏差修正。则有：

$$\boldsymbol{v}_{dw}^{c}=\frac{\boldsymbol{v}_{dw}}{1-\beta_1^t} \tag{3-29}$$

$$\boldsymbol{v}_{db}^{c}=\frac{\boldsymbol{v}_{db}}{1-\beta_1^t} \tag{3-30}$$

$$\boldsymbol{s}_{dw}^{c}=\frac{\boldsymbol{s}_{dw}}{1-\beta_2^t} \tag{3-31}$$

$$\boldsymbol{s}_{db}^{c}=\frac{\boldsymbol{s}_{db}}{1-\beta_2^t} \tag{3-32}$$

通过以上公式，可对第t轮迭代过程中参数梯度累积量值进行校正。可以基于动量和 RMSProp 算法的结合来实现权重值和偏置值的更新。

$$W=W-\alpha\frac{\boldsymbol{v}_{dw}^{c}}{\sqrt{\boldsymbol{s}_{dw}^{c}}+\varepsilon} \tag{3-33}$$

$$b=b-\alpha\frac{\boldsymbol{v}_{db}^{c}}{\sqrt{\boldsymbol{s}_{db}^{c}}+\varepsilon} \tag{3-34}$$

以上即为动量算法和 RMSProp 算法进行结合所形成的 Adam 算法，结合了两者的 Adam 优化算法有诸多优点：其计算高效，对内存要求较小，适合解决大规模数据和参数优化问题，可对高噪声或稀疏梯度的问题进行处理，同时其适应非稳态目标，超参数可直观解释，基本不需要调参，可为不同计算自适应学习率。鉴于 Adam 诸多优点，因此在诗歌生成模型的训练中，选择了 Adam 优化算法。

4 实验与分析

本研究主要使用 Chinese-poetry 诗歌数据库及部分其他网络诗歌数据集进行训练及实验，本章将首先介绍实验数据集及其预处理，然后对模型训练进行了介绍，接着通过与现有主流的绝句生成方法进行了比较实验，通过自动评估及人工评估对 KTEQG 模型进行验证，最后对模型进行了图灵测试。

4.1 数据集

本研究收集了来自 Chinese-poetry 诗歌数据库及部分其他网络诗歌数据集共计 376241 首诗歌[52]。绝句是中国传统诗歌中具有代表性的诗歌体裁，因此本研究主要通过绝句诗这一体裁诗歌进行实验，因考虑到绝句在唐朝形成了规范的章法格律要求，为避免过于久远绝句不够规范而带来的噪声影响，本研究从中筛选出唐朝及唐朝之后绝句诗 77824 首，首先从中随机抽取 2000 首绝句用作测试集，随机抽取 2000 首绝句用作验证集，剩下的作为模型训练集。

收集的原始绝句语料，需进行语料处理，才能作为训练语料用于绝句生成模型的训练，训练预料处理过程如下：首先需要对绝句诗歌进行分词，然后通过使用 2.3 节的主题提取方法进行关键词提取，即通过 TextRank 算法从每一句绝句中提取出一个关键词作为该句诗歌主题词，然后将每一句绝句的主题词和该绝句历史文本及当前生成文本组合成三元组，从而得到相关训练语料。如表 4-1，杜甫的这首绝句，可通过处理得到 4 条训练语料。

表 4-1 诗歌生成模型训练语料示例

主题词	历史生成诗歌文本	生成当前文本
黄鹂	—	两个黄鹂鸣翠柳
白鹭	两个黄鹂鸣翠柳	一行白鹭上青天
雪	两个黄鹂鸣翠柳，一行白鹭上青天	窗含西岭千秋雪
船	两个黄鹂鸣翠柳，一行白鹭上青天，窗含西岭千秋雪	门泊东吴万里船

4.2 模型训练

绝句数据的预处理及模型训练使用 Python 3.6 编程语言，在 TensorFlow 中进行绝句生成实现，CPU 为 INTER i74790k，内存大小为 128G，由于绝句数据量大，训练过程中使用 CPU 进行需要花费大量时间，因此实验选择了双路 NVI

-DIA GeForce GTX 1080Ti 进行。

在绝句生成阶段，模型存在两个编码器，其参数可以独立也可共享，因输入内容为绝句的历史生成和主题关键词，其在编码过程中存在许多共性，共享编码器参数可取得更好的生成效果，因此模型在绝句生成阶段选择使用共享编码器参数。具体的模型训练，通过从语料库中选取出现概率排名前 8000 的字构建输入输出词典，使用 3.1 节中介绍的 Word2vec 技术对词向量进行初始化，大小设定为 512,；编码器和解码器隐藏层大小设定为 512，Mini-Batch 设定为 128，基于 3.3 节中对于优化算法的讨论，我们此处选择了 Adam 算法[41]进行训练。

4.3 模型对比实验

为验证本研究提出的 KTEQG 模型在实际绝句生成中的表现，本文采用了 2.4 节中介绍的自动评估和人工评估方法进行诗歌评估，选取了目前表现较好的基于规划的诗歌生成模型（Planing based Poetry Generation,PPG）[17]和基于显著性上下文机制（Salient-Clue Mechanism，SCM）的诗歌生成模型[18]进行对比实验。对比实验过程中，使用相同语料库及预处理。

4.3.1 自动评估

自动评估使用 BLEU 算法进行，由 2.4 节对 BLEU 算法的相关介绍可知，此处评估绝句的最大困难是获得参考诗句。绝句生成过程中，给定相同的关键词和第一句话，人类产生的诗歌可能会非常多样化，因此需要具有不同写作风格和特征的诗句来确保评价质量。本文借鉴 He 等人的方法[12]，基于以下思想进行：如果两个诗句有相似的关键字，则其下一句可互为参考。 例如，“月明花满地”和“明月临沧海”关键词“明”和“月”，“明月花满地”下一句是“君自忆山阴”；“明月临沧海”下一句是“闲云恋故山”，而“君自忆山阴”和“闲云恋故山”表达了类似的内容和情感。所以“君自忆山阴”可以作为“闲云恋故山”的参考[18]。

假设给定诗句 S1，我们提取几个关键字词 A，B 和 C，并由《诗学含英》[19]中同一目录的相关关键字扩展组成关键字集，假设{A，A1，A2，B，B1，...，C，C1，C2，C3，...}代表该诗句的意思，如果另一个诗句 S2 具有关键字 A1，B1 和 C3，则 S1 和 S2 是相似的，S1 和 S2 的下一个诗句是彼此的参考。通过这种方式，可以从数据集中获得相关测试集。因绝句生成为五言和七言，其长度相等，且绝句诗基本元素为字或二字词语，所以 BP 值始终为 BLEU 指标统计到 bi-gram（即 2-gram）。

分别让三个模型生成相同主题的绝句 12 首，然后使用 BLEU 自动评估方法进行评估，结果如表 4-2 所示。

表 4-2 BLEU 评估方法试验结果

绝句生成模型	BLEU 评分
PPG	0.191
SCM	0.213
KTEQG	0.229

由实验结果可知，以上三种绝句生成方法评估结果相近，整体 BLEU 评分都很低，这是由于绝句生成不太可能产生与训练诗歌预料完全相同的诗。其中本文提出的 KTEQG 模型与 SCM 模型评分相近，略高于 SCM 模型，其中，由于 SCM 模型对诗歌风格进行了限制，实验过程中，若不对 SCM 模型进行绝句风格限制，该模型评分将产生较大波动。而相比 PPG 模型，KTEQG 模型具有明显优势。由此推断本研究提出的基于关键词转换扩展的绝句生成方法，生成的绝句具有更好的流畅性。

4.3.2 人工评估

本文对绝句的人工评估，基于 2.4 节的相关讨论，在对“流畅性”、“连贯性”、“格律性”、“意义”四个评价标准进行评估的同时，结合绝句的文学特点，关注了生成绝句的意境/情感和切题方面的评价，在原有基础上加入了“意境/情感”和“切题”两个评价标准，从而进行更为全面合理的人工评估。相关评估标准如表 4-3 所示。

表 4-3 绝句人工评估标准

评价标准	说明	评分标准
流畅性	生成诗歌的诗句是否自然流畅？	1-5 分
连贯性	生成诗歌的诗句之间是否前后关联？	1-5 分
格律性	生成的诗歌诗句是否满足古诗的格律要求？	1-5 分
意义	生成的诗歌是否真实表达内容及具体实际意义？	1-5 分
意境/情感	生成的诗歌是否具有情感和诗歌意境？	1-5 分
切题	生成的诗歌是否切合写作主题？	1-5 分

具体实验评估中，让三个模型分别生成 12 首绝句，然后由 5 位具有诗词文学专业背景的评审，参照表 4-3 的人工评估相关标准分别进行评分（评分精确到 0.1），然后取平均值进行统计，得到最终模型评分。其结果如表 4-4 所示：

表 4-4 人工评估结果

生成模型	流畅性	连贯性	格律性	意义	意境/情感	切题	平均
PPG	4.14	4.15	4.03	4.09	3.83	4.13	4.06
SCM	4.38	4.45	4.19	4.02	3.79	3.92	4.13
KETQG	4.31	4.38	4.34	4.14	3.87	4.21	4.21

由实验结果可以看出，在“格律性”、“意义”、“切题”三个指标上，本文提出的 KTEQG 模型评分明显高于另外两个模型，说明 KTEQG 模型通过用户意图确定唯一关键词，再进行文言文关键词转换扩展，然后基于注意力机制的编码器-解码器模型进行绝句生成，可以更好的处理诗句的平仄押韵，且在内容表达和主题表达上具有更好的表现；在“流畅性”“连贯性”两个指标中，KTEQG 模型评分和 SCM 模型相近，略低于 SCM 模型，说明在流畅性及上下文连贯性上，SCM 模型通过提取每一句已生成诗句显著性词语进行下句生成的方法可以带来更好的效果，KTEQG 模型整体效果也十分接近；在“意境/情感”这一指标上，三个模型评分都不高且分数相近，说明现阶段诗歌自动生成对生成诗句的情感和意境方面的把控还欠缺，还有进步空间；在整体平均评分上，本文提出的 KTEQG 模型取得了更好的评分，说明本文提出的 KTEQG 模型在绝句生成中，具有更好的生成效果。

4.4 模型图灵测试

图灵测试作为验证计算机是否拥有人类智能的一种方法，在绝句生成研究中，通过设计图灵测试，以证明本文提出的绝句生成模型达到了普通人类的创作水平，该实验对于评估绝句生成好坏具有可行性及参考价值。本研究中，将人类创作的绝句与本文提出的 KTEQG 模型生成的绝句进行对比，设计了两组图灵测试，一组为诗人组，诗人组绝句全部来自测试集，为诗人所创作，一组为大众组，大众组绝句为具有本科及以上学历文科背景的人所创作，这样设置的目的是为了将 KTEQG 模型同时与具有专业诗歌创作水平的人及无相关背景的普通大众进行比较，验证模型的智能程度。此处基于 2.4 节中介绍的图灵测试方法，进行了两组实验。

实验一：与诗人组创作的绝句对比实验。

实验设置评测人员 50 人，其中 6 人为专家评测组，具有汉语言古诗词方向专业背景，其他 44 人为普通评测组，具有本科及以上学历。诗人组从绝句数据集中的测试集中随机抽取 15 首诗歌，然后对抽取绝句进行主题概括，得到主题词作为输入，由 KTEQG 模型生成 15 首绝句，由此得到同一主题要求的绝句各 15 首，将 30 首绝句按同主题组成 15 张测试卡片，卡片中两首绝句，随机标记

为绝句一和绝句二，如图 4-1 示列。

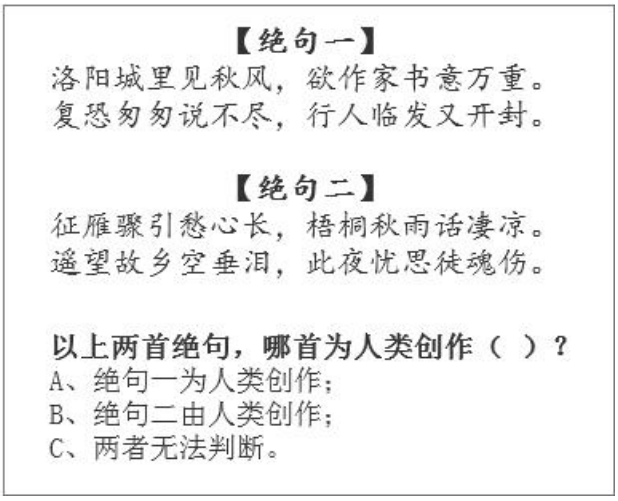

图 4-1 基于关键词转换扩展的绝句生成模型图灵测试示列

评测人员依次对 15 张卡片内容进行判断，从以下三个选项中进行选择：

A、绝句一为人类创作；

B、绝句二由人类创作；

C、两者无法判断。

对测试结果进行统计时，若把计算机生成的绝句诗判断为人类创作的，则记为错误，否则记为正确。

对测试结果进行统计，得到实验结果如表 4-5 所示：

表 4-5 KTEQG 模型创作绝句与诗人组创作绝句对比实验结果

	错误数（人/次）	错误率	无法判断数（人/次）	无法判断率	正确数（人/次）	正确率
专家评测组（共 75 人/次）	5	6.67%	7	9.33%	63	84.00%
普通评测组（共 675 人/次）	274	40.59%	61	9.04%	340	50.37%

实验二：与大众组创作的绝句对比实验。

与实验一不同的是，在与大众组创作的绝句对比实验中，首先让具有文科背景学历的本科及以上普通大众创作 15 首绝句诗，然后与 KTEQG 模型生成的绝句进行对比图灵实验。后续实验步骤和实验一相同，即将大众组创作的 15 首绝句提取主题词作为 KTEQG 模型输入，生成对应主题的 15 首绝句，组成 15 张测试卡片，再由评测人员对这 15 组绝句进行评估判断。评测人员配置、实验过程及统计方式和实验一相同，在此不再赘述。实验结果如表 4-6 所示。

表 4-6 KTEQG 模型创作绝句与大众组创作绝句对比实验结果

	错误数（人/次）	错误率	无法判断数（人/次）	无法判断率	正确数（人/次）	正确率
专家评测组（共 75 人/次）	17	22.67%	9	12.00%	49	65.33%
普通评测组（共 675 人/次）	362	53.63%	74	10.96%	239	35.41%

实验结果分析：

（1）在与诗人组创作的绝句对比实验中，普通评测组错误率达到了 40.59%，即 40.59%的普通评测组实验人员将计算机创作的绝句错判成了人类创作的绝句，另外有 9.04%的绝句选择无法判断，判断错误率超过图灵测试 30%基准，但在专家评测组中错误率只有 6.67%。由此可以得出，一般人是难以区分 KTEQG 模型生成的绝句和专业诗人创作的绝句，但对于具有诗歌相关专业背景的专业评测人员来说，还是可以明显区分。

（2）在与大众组创作的绝句对比实验中，KTEQG 模型生成的绝句被普通评测组错误判断为人类创作的绝句的比例高达 53.63%，同时还有 10.96%的绝句无法判定，结果远超图灵测试 30%基准线，说明本文提出的 KTEQG 绝句生成方法完全具有普通人所具有的创作水平；而专家评测组错误率达到 22.67%，同时无法判断率 12.00%，相比实验一，专家评测组错误率和无法判断率都有明显提升，虽然错误率未超过 30%，但加上无法判断率也已经超过 30%。

由此可推断出，本文提出的 KTEQG 绝句生成模型生成的绝句水平完全可以达到一般普通大众创作的诗歌水平，对于具有诗词背景的专业人士，也具有一定的迷惑性，接近了专业诗人水平。

5 总结与展望

5.1 工作总结

绝句是中国传统诗歌中具有代表性的诗歌体裁，是一种重要而特殊的文化遗产，普通人想要创作一首合格的绝句不容易，对于绝句等体裁的诗歌自动生成的研究，有利于中华传统文化的传播、促进中国传统诗歌的创新与发展、启发其他文本类型的生成研究，促进自然语言处理相关技术的发展。本文针对绝句生成中普遍存在的主题漂移、语义不连贯等问题，基于深度学习技术进行了绝句生成的深入研究。

本文提出了一种基于关键词转换扩展的绝句生成模型（Keyword Transforma-tion and Expansion Quatrain Generation Model，KTEQG），该模型将绝句生成分为关键词转换、关键词扩展和绝句生成三个阶段。首先对包含用户写作意图的文本进行唯一关键词提取，再进行关键词文言字词转换，然后基于转换后的主题关键词进行扩展，为待生成的每一句绝句分配主题关键词，最后基于注意力机制的编码器-解码器将分配的主题关键词和历史生成的内容作为输入进行绝句生成。同时针对本文提出的绝句生成模型，进行了自动评估、人工评估和图灵测试：在自动评估中，使用 BLEU 算法，让 KTEQG 模型与现有主流生成模型 PPG 和 SCM 模型比较实验，取得了不错的 BLEU 评分，说明了本文提出的 KTEQG 模型生成的绝句在流畅性上有较好的表现。在人工评估中，针对现有的诗歌评价方法对生成诗歌切题、意境和情感等方面指标欠缺的问题，本文结合绝句相关文学特点，对原有的人工评价方法进行了完善。通过与 PPG 和 SCM 模型对比实验，结果显示 KTEQG 模型在“格律性”、“意义”、“切题”三个指标上得分明显高于其他两个模型，说明 KTEQG 模型通过用户意图确定唯一关键词，再进行文言文关键词转换扩展，然后基于注意力机制的编码器-解码器模型进行绝句生成这一方法，可以更好的处理诗句的平仄押韵，且在内容表达和主题表达上具有更好的表现。

在图灵测试中，通过分别与专业诗人组、普通大众组进行图灵实验，实验结果表明本文提出的 KTEQG 模型生成的绝句水平完全可以达到一般普通大众的诗歌创作水平；而与具有诗词专业背景的人进行图灵测试，也具有一定的迷惑性，接近了专业诗人水平。

5.2 研究展望

本文提出的基于关键词转换扩展的绝句生成模型，取得了不错的实验结果。但在研究的过程中，也产生了许多思考：

（1）本文对于绝句生成方法的研究主要是基于绝句这一诗歌体裁，因选取的诗歌预料主要为唐代及以后的绝句诗，其格律等相对规范，因此 KTEQG 模型在绝句生成中表现优异，但中国传统诗歌体裁丰富，不同的诗歌体裁有不同的格律等要求，如何让模型更具迁移能力，让 KTEQG 模型可以简单方便的应用到其他体裁诗歌生成，甚至是到其他文本生成任务中使用，值得深入研究。

（2）现有的绝句等类型诗歌生成方法研究，主要是针对诗句的研究，对于诗歌题目的生成研究是空白的。一首完整的古诗，具有一个表达诗句主题的题目是必不可少的，对于诗歌题目自动生成的研究同样是自然语言处理领域具有挑战性的课题，如何生成带诗歌题目的完整诗歌生成值得探究。

（3）虽然本文针对现有绝句等诗歌生成模型的评估方法进行了总结完善，但现有的诗歌生成模型评估主要还是依赖于人工，这就使得评估结果带有主观性。因此建立更加完善科学的诗歌评估体系很有必要，特别是诗歌自动评估方法的研究。

总之，绝句等诗歌自动生成方法的研究，还存在许多值得探讨的问题，需要我们不断发现，不断深入研究。

参考文献

[1] Krizhevsky A, Sutskever I , Hinton G . ImageNet Classification with Deep Convolutional Neural Networks[J]. Advances in neural information processing systems, 2012, 25(2).

[2] 梅敬忠. 唐诗宋词赏析—中国古典诗歌的鉴赏艺术[J]. 领导科学论坛, 2018(16):63-78.

[3] 冯佳宁. 唐绝句章法艺术研究[D]. 南京师范大学, 2018.

[4] Wang L. A summary of rhyming constraints of chinese poems[M]. [S.1.]:Beijing Press, 2002.

[5] 李星宇, 王丽娟. 基于古诗文知识图谱的诗词创作系统[J], 计算机产品与流通, 2019(04):106.

[6] Boden M A. Creative Mind:Myths and Mechanisms[J]. Begavioral & Brain Sciences, 1994, 17(3):519-531.

[7] Hartman C O. Virtual Muse: Experiments in Computer Poetry[M]. [S.I.]: Wesleyan University Press, 1996.

[8] Gerv' as P. Wasp:Evaluation of different strategies for the automatic generation of spanish verse[C]//Proceedings ofthe AI SB-00 symposium on creative&cultural aspects of AI. [S.1], 2000:93-100.

[9] Belén Díaz-agudo, Pablo Gervás, Pedro A. González-calero. Poetry Generation in COLIBRI.[M]// Proceedings of the Seventh Japan Congress on Testing Materials /. The Society of Materials Science, Japan, 2002.

[10] 周昌乐, 游维, 丁晓君. 一种宋词自动生成的遗传算法及其机器实现[J]. 软件学报, 2010, 021(003):427-437.

[11] Rui Yan, Han Jiang, Mirella Lapata. i, Poet: Automatic Chinese Poetry Composition through a Generative Summarization Framework under Constrained Optimization[C]// International Joint Conference on Artificial Intelligence. 2013.

[12] He J, Zhou M, Jiang L. Generating chinese classical poems with satistical machine translation models[C], AAAL 2012.

[13] Zhang X, Lapata M, Chinese poetry generation with recurrent neural networks[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing(EMNLP). Doha, Qatar:Association for

Computational Linguistics, October 2014:670-680.

[14] Wang Q , Luo T , Wang D , et al. Chinese Song Iambics Generation with Neural Attention-based Model[J]. 2016.

[15] Yan R. i, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema[C], IJCAI. 2016:2238-2244.

[16] Ghazvininejad M, Shi X, Choi Y, et al. Generating Topical Poetry[C], Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing．2016:1183-1191.

[17] Wang Zhe, He Wei, Wu Hua, et al. Chinese Poetry Generation with Planning based Neural Network[J]. 2016.

[18] Xiaoyuan Yi, Ruoyu Li and Maosong Sun.2018. Chinese Poetry Generation with a Salient-Clue Mechanism. In Proceedings of CoNLL 2018.

[19] 刘文蔚．诗学含英[M]．香港银行出版社．2001.

[20] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. Computer Science, 2014.

[21] Mikolov T , Martin Karafiát, Burget L , et al. Recurrent neural network based language model[C]// INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association,Makuhari, Chiba, Japan, September 26-30, 2010. DBLP, 2010.

[22] 赵京胜，朱巧明，周国栋，张丽．自动关键词抽取研究综述[J]．软件学报，2017，28(09):2431-2449.

[23] Jones K S . Jones, K.S.: A Statistical Interpretation of Term Specificity and its Application in Retrieval. Journal of Documentation 28(1), 11-21[J]. Journal of Documentation, 1972, 28(1):11-21.

[24] BLEI D M, NGAY, JORDAN M I. Latent dirichlet allocation [J], Journal of Machine Learning Research, 2003, 3:993-1022.

[25] MIHALCEA R, TARAU P. Textrank:brinng order into text [C]. Pr-oceedings of the 2004 Conference on Empirical Methods in NaturalLanguage Processing. Association for Computational Linguistics, 2004:404-411.

[26] 曹洋．基于 TextRank 算法的单文档自动文摘研究[D]．南京大学，2016.

[27] 百度．百度翻译开放平台[EB/OL]． http://api.fanyi.baidu.com/doc/11.

[28] 徐戈，王厚峰．自然语言处理中主题模型的发展[J]．计算机学报，2011, 34(08):1423-1436.

[29] 杜清运，任福．空间信息的自然语言表达模型[J]．武汉大学学报(信息科学版)，2014，39(06):682-688.

[30] Chen S F, Goodman J. An empirical study of smoothing techniques for language modeling[C]//Joshi AK, Palmer M, 34th Annual Meeting of the Association for Computational Linguistics, 24-27 June 1996, University of California, Santa Cruz, California, USA，Proceedings. [S.1]: Morgan Kaufmann Publishers/ACL, 1996: 310-318.

[31] Bengio Y, Ducharme R, Vincent P'et a1. A neural probabilistic language model[J]. Journal ofMachine Learning Research. 2000. 3:1137-1155.

[32] 王哲．基于深度学习技术的中国传统诗歌生成方法研究[D]．中国科学技术大学，2017.

[33] Chung J , Gulcehre C , Cho K , et al. Gated Feedback Recurrent Neural Networks[J]. Computer ence, 2015:2067-2075.

[34] K. Papineni, S. Roukos, T. Ward, W. Zhu. BLEU: a method for automatic evaluation of MT. IBM research division, T J Watson Research Centre, Research Report: Computer Science RC22176 (W0109-022), 2001.

[35] 王改娣．论诗歌评价的标准: 从柏拉图到朱光潜[J]．英美文学研究论丛，2019(02):311-321.

[36] Turing A M. Computing Machinery and Intelligence[J]. Mind, 1950, 59(236):433-460.

[37] 万赟．从图灵测试到深度学习:人工智能 60 年[J]．科技导报，2016, 34(07):26-33.

[38] 李丹．基于长短时记忆网络的中文文本情感分析[D]．北京邮电大学, 2017.

[39] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality[J]. Advances in Neural Information Processing Systems, 2013,26:3111-3119.

[40] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013.

[41] Miller G A , Richard B , Christiane F , et al. Introduction to WordNet: An On-Line Lexical Database[J]. International Journal of Lexicography, 1991, 3(4):235--244.

[42] Thushan Ganegedara. Natural language processing with TensorFlow. [J].2018

[43] LeCun Y, Boser B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition[J]. Neural computation, 1989, 1(4): 541-551.

[44] Nal Kalchbrenner, P.Blunsom. Recurrent continuous Translation Models[c]//. EMNLP) 2013, 2013(1): 1700-1709.

[45] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation.

1997, 9(8): 1735-1780.

[46] Kim Y. Convolutional Neural Networks for Sentence Classification[J]. Eprint Arxiv, 2014.

[47] Léon Bottou. Stochastic Gradient Descent Tricks[J]. 2012.

[48] Palm R B. Deep learning toolbox[EB/OL]. (2015-09). http://www.mathw-ork.com/mathlabcentral/filees-change/38310-deep-learning-toolbox.

[49] Qian N. On the momentum term in gradient descent learning algori-thms[J]. Neural Networks, 1999, 12(1):145-151.

[50] Ruder S. An overview of gradient descent optimization algorithms[J]. arXiv preprint arXiv:1609. 04747, 2016.

[51] Kingma D , Ba J . Adam: A Method for Stochastic Optimization[J]. Computer ence, 2014.

[52] Gaojunqi. Chinese-poetry: 最全中文诗歌古典文集数据库[EB/OL]. https://github.com/chinese-poetry/chinese-poetry, 2019-9-13.

致　谢

时光飞逝，八年前踏入师大校园的青涩少年，终于迎来了告别。师大八年，遇见了许多，学习了许多，成长了许多。

首先想要感谢我的导师龚俊老师，本论文的完成，离不开老师的悉心指导。研究生三年，老师严于律己、宽以待人的崇高风范及平易近人的人格魅力对我产生了深刻的影响，从中学到的不仅仅是学术知识，还有更重要的为人处世的方式方法。然后想要感谢对本论文提供帮助的朋友们，你们的帮助让我更顺利的完成了相关研究。

感谢师大八年遇见的所有美好：本科六栋那伙纯粹的兄弟们、社团里可爱的“蓝精灵”们、大活三楼并肩“战斗”的兄弟姐妹们、贵州望谟那些可爱的老师和孩子们、亲爱的支教队友们、研究生阶段一起打球一起喝酒的兄弟们、以及这八年遇到的所有挫折与美好……这八年有太多的相遇，也许有好有坏有喜有悲，但回头看都成为了最美好的记忆，感谢相遇。

最后感谢我的家人，感谢你们在我二十年的学习生涯里最无私的支持与爱。

感谢遇到的每一个人。

在读期间公开发表论文（著）及科研情况

一、发表论文情况

[1] Hanyu Liu. Research on Intelligent Writing Poetry Model Based on Neural Networ[J]. Academic Journal of Computing & Information Science. 2019, 48(08):36-41.