文章编号：2096-1472(2022)-11-23-05
DOI:10.19644/j.cnki.issn2096-1472.2022.011.006

基于检索与生成混合模型的个性化聊天机器人系统的设计与实现

孔繁恒，高永祺，张子帅，阮俊豪，冯 时

(东北大学计算机科学与工程学院，辽宁 沈阳 110819)
✉kongfanheng426@163.com; 905323742@qq.com; 1060876095@qq.com;
954369920@qq.com; fengshi@stu.neu.edu.cn

摘 要：聊天机器人依据其对话生成技术主要分为检索模型和生成模型，但这两种模型各有优劣。通过设计一种混合检索与生成的混合对话模型，保留了两种模型的优点并尽量避免其缺点。混合模型既具有检索模型的信息性，又具有生成模型的多样性，人机对话更加流畅、生动。使用Elasticsearch搜索服务器大大降低了系统回复生成的时间，回复速度提高了10%。此外，考虑到聊天机器人在对话过程中难以保持一致的个性，采用PostKS(Posterior Knowledge Selection)模型嵌入个性属性，实现了个性化的智能聊天机器人系统。文中提出的混合模型实现了个性化的回复且具有较快的回复速度，更加符合用户的需求。

关键词：聊天机器人；Elasticsearch；PostKS模型；个性化
中图分类号：TP311.1 **文献标识码**：A

Design and Implementation of Personalized Chatbot System based on Retrieval and Generation Hybrid Model

KONG Fanheng, GAO Yongqi, ZHANG Zishuai, RUAN Junhao, FENG Shi

(*School of Computer Science and Engineering*, *Northeastern University*, *Shenyang* 110819, *China*)
✉kongfanheng426@163.com; 905323742@qq.com; 1060876095@qq.com;
954369920@qq.com; fengshi@stu.neu.edu.cn

Abstract: According to their dialogue generation technology, chatbots can be mainly divided into retrieval model and generation model, which have both strengths and weaknesses. This paper proposes to design a hybrid dialogue model of retrieval and generation, to retain the strengths of the two models while avoiding as many weaknesses as possible. The hybrid model not only has the information of retrieval model but also has the diversity of generating model, so the man-machine dialogue is more smooth and vivid. Using the Elasticsearch search server greatly reduces the system reply generation time and the response speed is increased by 10%. In addition, considering that it is difficult for chatbots to maintain consistent personalities in the process of dialogue, PostKS (Posterior Knowledge Selection) model is used to embed personality attributes to realize the personalized intelligent chatbot system. The proposed hybrid modelr realizes personalized reply and has a faster reply speed, which is more in line with the needs of users.

Keywords: chatbot; Elasticsearch; PostKS model; personalization

1 引言(Introduction)

伴随着互联网数据的爆发式增长，聊天机器人产品出现在人们的视野中。聊天机器人是经由对话或文字进行交谈的计算机程序，能够模拟人类对话，为用户提供了智能、自然连贯的对话服务。聊天机器人根据其对话生成技术主要分为检索式[1−3]和生成式[4]。检索式模型使用信息检索的技术，从事先处理好的语料库中选择匹配的对话作为应答。生成式模型使用深度学习的技术，通过预先训练的模型合成适当的应答。

在对话中保持一致的个性对于人类来说是很自然的，但对于机器来说却不是一件平凡的任务，由于在自然语言中体现个性的困难，以及在大多数对话语料库中观察到的人物角色稀疏问题，仍然没有得到很好的探讨。本文设计了一种检索−生成混合模型[5−6]，该模型可以利用人物角色稀疏对话数据生成连贯的响应。通过对说话人的角色和对话历史进行编码，设计个性属性嵌入，来模拟更丰富的对话语境，从而实现个性化的聊天机器人系统。

基金项目：国家自然科学基金面上项目(61872074).

2 需求分析(Requirements analysis)

作为人机交互领域一个非常重要的研究方向，人机对话系统正处于蓬勃发展的阶段。传统的问答系统已经不能满足人们的要求，如今人们更偏向于系统拥有更广泛的知识领域并能够具有特定的个性，使得人机对话更加生动、顺畅。聊天机器人，主要任务是利用自然语言完成与人在任意话题上的交流，其涉及的话题之广、表达方式之多更能符合用户的需求。

本系统所要实现的目标为个性化的智能聊天机器人系统，拥有进行个性化聊天的功能。用户在与本系统交互时，可以设置聊天机器人的个性信息。系统会通过对用户设置的角色和对话历史进行编码，模拟对话语境，分别得到检索式和生成式两种回复方式，将两者评价比较后选择最优回复，并显示在聊天界面上，实现个性化的聊天。本系统的各功能模块如图1所示。

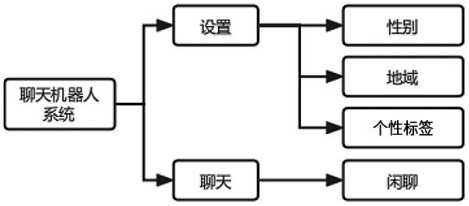

图1 系统功能模块图

Fig.1 System function block diagram

3 系统设计和实现(System design and implementation)

3.1 总体设计

本系统以PyTorch为开发框架，采用深度学习的技术，使用检索-生成混合模型，构建了一个个性化的智能聊天机器人系统。前端采用Tkinter模块接口，后端采用PyTorch开发框架，均使用Python语言编写。前端为Graphical User Interface(GUI)，分为两部分：设置界面(可供用户设置聊天机器人的个性信息，包括性别、地域和个性标签)与聊天界面。后端使用检索-生成混合模型，检索模型采用Elasticsearch搜索服务器[7]，检索速度极快，大大降低了整体系统的回复生成时间；生成模型采用PostKS模型[8]，可以定向设置聊天机器人的个性并生成具有特定个性的回复。通过混合两种模型提高了回复的相关性和流畅性，语句质量较高。

在后端设计中，选择具有个性信息的2019 年社会媒体处理大会中文人机对话技术评测(SMP2019-ECDT)对话集和质量较高的小黄鸡语料库[9]作为数据集，然后进行数据预处理，包括筛选高频个性标签、数据清洗、调整数据格式等。检索模块使用Elasticsearch搜索服务器及小黄鸡语料库构建检索模型，生成模块首先构建PostKS个性化生成模型，然后利用SMP2019-ECDT对话集进行训练。前端为用户提供窗口输入个性信息与对话，以字符串的形式传送到后端。后端接收到用户发送的信息，分别传送到检索和生成模块中生成应答，在回复选择模块比较两种模型生成的应答并选择合适的应答，同时将此应答传回前端并展示到界面上。系统总体结构如图2所示。

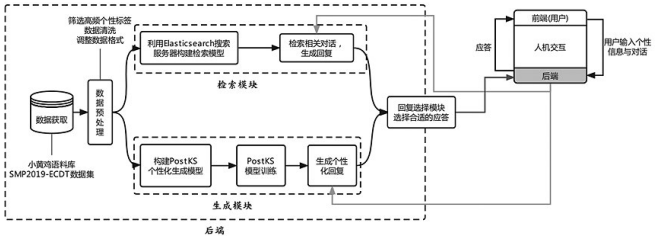

图2 系统总体结构图

Fig.2 System overall structure diagram

3.2 数据集获取及预处理

本文选取SMP2019-ECDT提供的数据集作为生成模型的原始数据，数据集中包含约500万轮次的对话(既包含单轮对话又包含多轮对话)，以及参与这些对话的发话人的个性化信息(包括性别、个性标签、所属地域)。数据集为JSON格式，如{"dialog"：[["疯了疯了"]，["我要带到教室去上课你就羡慕我吧！"]]，"profile"：[{"tag"：["美食"]，"loc"：“四川 成都”，"gender"："female"}，{"tag"：["娱乐;旅游"]，"loc"："四川"，"gender"："female"}]，"uid"：[0，1]}，其中dialog为对话，profile表示对话人的个性化信息，包括tag——个性标签、loc——所属地域、gender——性别、uid——对话人id。

由于生成模型需要个性信息作为特征，标签的筛选十分重要，因此要对数据集进行预处理。首先将相似的标签进行合并，例如“电影”“爱电影”“电影爱好者”这类标签可以合并成一类。然后筛选出高频出现的标签，得到高频标签字典。算法描述如下所示。

```
# 输入：初始数据集D
# 输出：vocab_tag为高频标签字典
vocab_tag={}
for each data in D：
    #将每轮对话双方的标签分解成单个标签
    tag_list=split tags_string to list
    #统计每个标签出现的次数
    for tag in tag_list：
        if tag not in vocab_tag：
            vocab_tag[tag]=1
        else：
            vocab_tag[tag]+=1
#取出高频tag
vocab_tag=dict((k，v) for k，v in vocab_tag.items() if v>=1000)
return vocab_tag
```

由于数据集中存在很多个性信息不完善(缺少性别/地域/个性标签)的对话，因此需要对数据集进行清洗，并且保证对话中的每个单词都在单词表中，以此来提升数据集的质量。

算法描述如下所示：

```
#输入：初始数据集D，高频标签字典vocab_tag，单词表V
#输出：清洗后的数据集C
for each data in D：
    #没有个性标签/地域/性别信息
    if tag is None or loc is None or gender is None：
        continue
    for each tag in tags：
        #统计高频标签的个数
        if tag in vocab_tag：
        count=count+1
    #个性信息充足且对话中每个单词都在单词表中
    if count>=3 and every word in V：
        add data to C
return C
```

对于检索模型，SMP2019-ECDT提供的数据集对话噪音较多，在语意连贯、流畅性等方面有所欠缺，因此本文采用小黄鸡语料库作为检索模型的数据集。该数据集只需调整数据格式即可使用。

3.3 检索模块

Elasticsearch是一个分布式、可扩展、实时的搜索与数据分析引擎。由于Elasticsearch是在Lucene基础上构建而成的，所以在全文本搜索方面表现十分出色。Elasticsearch是一个近实时的搜索平台，这意味着从文档索引操作到文档变为可搜索状态之间的延时很短，相比一些传统的检索方法回复速度提升了10%以上。Elasticsearch提供了强大的索引能力，其使用的倒排索引相较于关系型数据库的B-Tree索引更快。

通过文本匹配的相关度分数来判断检索对话的相关性。其相关度评分的计算规则涉及布尔模型(Boolean Model)、词频-逆文档频率(TF-IDF)和向量空间模型(Vector Space Model)。Elasticsearch使用布尔模型查找匹配文档，并用一个名为实用评分函数(借鉴TF-IDF和向量空间模型)的公式来计算相关度，同时也加入了协调因子、字段长度归一化、词或查询语句权重提升。

映射(Mapping)是定义文档及其包含的字段如何存储和索引的过程。每个文档都是字段的集合，映射数据时，将会创建一个映射定义，其中包含与文档相关的字段列表。

本文检索的映射规则设置如下：

```
mapping={
    'properties'：{
        'question'：{
            'type'：'text',
            'analyzer'：'ik_max_word',
            'search_analyzer'：'ik_smart'
        }
    }
}
```

其中，type表示每个字段的数据类型，常用的有keyword和text等，可根据检索的目的设置。analyzer表示索引时用于文本分析的分析器，search_analyzer表示搜索时使用的分析器，本文分别设置为ik_max_word和ik_smart。IK分词器是一款中文分词器，ik_max_word算法是最细粒度切分算法，比如将“中华人民共和国人民大会堂”拆分为中华人民共和国、中华人民、中华、华人、人民共和国、人民、共和国、大会堂、大会、会堂等词语。而ik_smart算法是最粗粒度切分算法，比如将“中华人民共和国人民大会堂”拆分为中华人民共和国、人民大会堂。对于中文内容，在索引时用ik_max_word，在搜索时用ik_smart可以使检索的效果达到最佳。

3.4 生成模块

生成模块中使用PostKS模型，这是一个可以利用知识的先验和后验分布来促进知识选择的神经模型。通过使用先验分布有效地逼近后验分布，该模型可以在推理过程中产生适当的回答。根据用户输入的个性化信息通过知识管理器来设定聊天机器人的个性，使得系统能够生成具有特定个性的回复。

假设一个语句$X=x_1x_2\cdots x_n$(x_t是X中第t个单词)，存在一个知识集合$\{K_i\}_{i=1}^N$，目标是从集合中选择适当的知识，并通过选择的知识生成应答$Y=y_1y_2\cdots y_n$。PostKS模型的架构如图3所示，主要由四部分组成。

(1)对话编码器(Utterance Encoder)：将X编码成一个对话向量x，并将其输入知识管理器。

(2)知识编码器(Knowledge Encoder)：将每一个K_i作为输入，然后把它编码成知识向量k_i，当回复Y可用时将其编码成向量y。

(3)知识管理器(Knowledge Manager)：由先验知识模块和后验知识模块两个子模块组成。给定先前编码的x和$\{K_i\}_{i=1}^N$(如果可用)，知识管理器负责选择一个适当的k_i并使它(与一个基于注意的上下文向量c_t)进入解码器。

(4)解码器(Decoder)：根据所选知识k_i以及基于注意的上下文向量c_t生成回复。

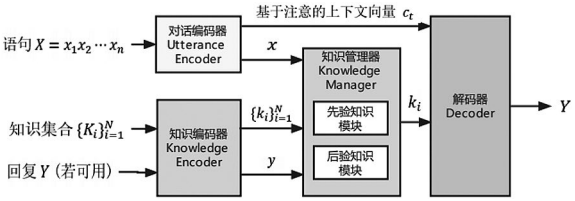

图3 PostKS模型图

Fig.3 PostKS model diagram

(1)对话编码器和知识编码器

在对话编码器中使用Gate Recurrent Unit(GRU)的双向Recurrent Neural Network(RNN)(由前向RNN和反向RNN组成)，对于语句$X=x_1x_2\cdots x_n$，前向RNN从左向右读取X并

获得每个x_t从左向右的隐藏状态$\overrightarrow{h_t}$，同理反向RNN从右向左读取X并获得每个x_t从右向左的隐藏状态$\overleftarrow{h_t}$，这两个隐藏状态合并成一个总的隐藏状态h_t：

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right] = \left[\mathrm{GRU}\left(x_t, \overrightarrow{h_{t-1}}\right); \mathrm{GRU}\left(x_t, \overleftarrow{h_{t-1}}\right)\right]$$

利用隐藏状态并定义$x = \left[\overrightarrow{h_T}; \overleftarrow{h_1}\right]$，这个向量会传送到知识管理器来优化知识的选择，同时它将会作为解码器的初始隐藏状态。

知识编码器的结构与对话管理器相同，但它们之间不共享参数。知识编码器使用双向RNN将每一个知识K_i(如果回复Y可用)转化成向量k_i，并传入知识管理器。

(2)知识管理器

给定编码后的语句x和知识集合$\{k_i\}_{i=1}^N$，知识管理器的目标是选择一个合适的个性k_i，当应答y可用时，模型也将同时利用y得到k_i。

在先验知识模块中，定义了知识上的条件概率分布$p(k|x)$：

$$p(k = k_i|x) = \frac{\exp\left(k_i \cdot x\right)}{\sum_{j=1}^{N} \exp\left(k_j \cdot x\right)}$$

使用点乘来衡量k_i与x的关联性，$p(k|x)$表示仅有的时候，即它在不知道应答的情况下工作，因此它是知识先验分布。但不同对话相关的知识各有不同，所以在训练中仅仅使用先验分布来选择知识是十分困难的。

在后验知识模块中，通过考虑输入语句及其应答，定义了知识的后验分布$p(k|x,y)$：

$$p(k = k_i|x,y) = \frac{\exp\left(k_i \cdot \mathrm{MLP}([x;y])\right)}{\sum_{j=1}^{N} \exp\left(k_j \cdot \mathrm{MLP}([x;y])\right)}$$

其中，MLP是一个全连接层。通过比较先验分布，后验分布是比较准确的，可以获得应答Y使用的知识内容。

显然，后验分布在选择知识时是优于先验分布的，但在推理生成应答阶段中，后验分布是未知的，因此期望先验分布能够尽可能地接近后验分布。为此，引入了KLDivLoss(Kullback−Leibler Divergence Loss)作为辅助损失函数，用来衡量先验分布和后验分布的接近性：

$$L_{KL}(\theta) = \sum_{i=1}^{N} p(k = k_i|x,y) \log \frac{p(k = k_i|x,y)}{p(k = k_i|x)}$$

其中，θ为模型参数。

(3)解码器

解码器通过合并所选知识k_i逐字生成应答，使用分级门控融合单元(Hierarchical Gated Fusion Unit，HGFU)。HGFU提供了一种将知识融合到应答生成中的方法，由话语GRU、知识GRU和融合单元三个主要部分组成。

话语GRU和知识GRU遵循标准GRU结构，基于上一个状态s_{t-1}和上下文向量c_t，分别对最后生成的y_t和选择的知识k_i生成隐藏状态：

$$s_t^y = \mathrm{GRU}(y_{t-1}, s_{t-1}, c_t)$$

$$s_t^k = \mathrm{GRU}(k_{t-1}, s_{t-1}, c_t)$$

然后，融合单元将它们组合在一起，生成整体的隐藏状态：

$$s_t = r \odot s_t^y + (1 - r) \odot s_t^k$$

其中，$r = \sigma(W_z[\tanh\left(W_y s_t^y\right); \tanh\left(W_k s_t^k\right)])$，$W_z$、$W_y$、$W_k$是参数。门控$r$控制$s_t^y$和$s_t^k$对最终隐藏状态$s_t$的影响比例，以便于它们能够灵活地融合。

获得隐藏状态s_t后，下一个单词y_t根据下面的概率分布生成：

$$y_t \sim p_t = \mathrm{softmax}(s_t, c_t)$$

3.5 回复选择模块

检索模型与生成模型各有利弊。检索模型直接使用语料库中的对话，选择匹配度较高的对话进行回复，回复语句的质量较高，保证了对话的信息性，但无法回复语料库中未出现的问题；生成模型将对话经过一系列处理从而产生相应的回答，且PostKS模型能够根据特定的个性产生对应个性的回复，具有多样性，但相较于检索模型产生的回复可能会出现缺乏相关度或语义不通等问题。

为使聊天机器人系统既能保持检索模型的信息性，也能保持生成模型的多样性，且尽可能地克服两种模型的缺点，因此设计了一种检索-生成混合模型，结构如图4所示。

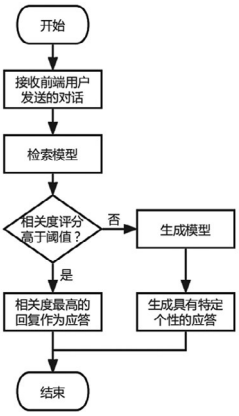

图4 回复选择模块流程图

Fig.4 Reply selection module flowchart

当接收到用户发送的对话时，先在检索模块通过Elasticsearch搜索引擎进行检索并进行相关度评分，设定一个适当的阈值(本文设为5)，若存在相关度评分高于阈值的对话，则将评分最高的回复发送给用户，否则调用PostKS个性化生成模型，依据用户设置的个性化信息，生成具有特定个性的回复发送给用户。

由于用户设置过聊天机器人的个性信息，当询问聊天机器人的性别与地域相关问题时，为防止机器人的回复与先前用户设置的信息不一致，机器人的回复不应通过检索或生成模型来获取，而是调用回复性别或地域的函数。

3.6 前端设计

前端使用Tkinter模块，是Python标准Tk GUI库。前端包括设置界面(可供用户设置聊天机器人的个性信息，包括性别、地域和个性标签)与聊天界面。

在设置界面(图5)分别输入性别、地域和个性标签即可设

置聊天机器人的个性，点击“确定”后进入聊天界面(图6)，在输入框中输入语句，点击“发送”即可生成应答并显示在界面中，点击“返回”将退回到设置界面，可重新设置聊天机器人的个性。

图5 设置界面

Fig.5 Interface for settings

图6 聊天界面

Fig.6 Interface for chatting

4 系统展示(System display)

系统运行结果如图7和图8所示。

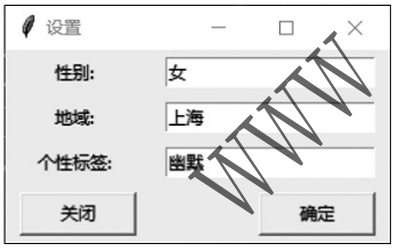

图7 设置界面展示

Fig.7 Display of setting interface

图8 聊天界面展示

Fig.8 Display of chatting interface

5 结论(Conclusion)

本文设计了一种可以实现个性化回复的聊天机器人系统。本系统基于PyTorch开发框架，后端各个模块结构清晰简洁，易于修改、扩展。使用检索-生成混合模型，既保留了检索模型的信息性，语句流畅、质量高，也保留了生成模型的多样性，且可生成特定个性的应答。检索模型使用Elasticsearch搜索引擎，检索速度极快，大大缩减了整个系统生成应答的时间，通常情况下1 s内即可产生应答，对话具有实时性。生成模型使用PostKS模型，可以生成具有特定个性的应答。本系统可供用户闲聊，同时也可以应用到多种不同的场景中。在面向特定的应用场景时，依据实际场景采取相应的个性，会使交流过程更为顺畅和专业，这无疑具有一定的社会价值。

参考文献(References)

[1] 吴俣,李舟军.检索式聊天机器人技术综述[J].计算机科学,2021,48(12):278-285.

[2] 赵芸,刘德喜,万常选,等.检索式自动问答研究综述[J].计算机学报,2021,44(06):1214-1232.

[3] GU J C, LI T D, LIU Q, et al. Speaker-Aware BERT for Multi-Turn Response Selection in Retrieval-Based Chatbots[C]// Association for Computing Machinery. International Conference on Information and Knowledge Management(CIKM). New York: ACM, 2020:2041-2044.

[4] 王艳秋,管浩言,张彤.聊天机器人的分类标准和评估标准综述[J].软件工程,2021,24(02):2-8.

[5] 杨慧敏,马廷淮.融合检索与生成的复合对话模型[J].计算机科学,2021,48(08):234-239.

[6] 张凉.基于检索与生成模型相结合的聊天机器人关键技术研究[D].上海:华东师范大学,2020.

[7] Elasticsearch. Elasticsearch Guide[EB/OL]. (2022-01-01)[2022-03-02]. https://www.elastic.co/guide/en/elasticsearch/reference/7.17/index.html.

[8] LIAN R Z, XIE M, WANG F, et al. Learning to Select Knowledge for Response Generation in Dialog Systems[C]// International Joint Conferences on Artificial Intelligence Organization. International Joint Conference on Artificial Intelligence(IJCAI). San Francisco: Morgan Kaufmann, 2019:5081-5087.

[9] candlewill.Dialog_Corpus[EB/OL].(2017-10-17)[2022-03-02]. https://github.com/candlewill/Dialog_Corpus.

作者简介：

孔繁恒(2001-)，男，本科生.研究领域：自然语言处理.

高永祺(2001-)，男，本科生.研究领域：自然语言处理.

张子帅(2000-)，男，本科生.研究领域：自然语言处理.

阮俊豪(2000-)，男，本科生.研究领域：自然语言处理.

冯　时(1981-)，男，博士，副教授.研究领域：数据挖掘，自然语言处理.