北京大学学报(自然科学版) 第 57 卷 第 1 期 2021 年 1 月
Acta Scientiarum Naturalium Universitatis Pekinensis, Vol. 57, No. 1 (Jan. 2021)
doi: 10.13209/j.0479-8023.2020.089

开放域对话系统的抗噪回复生成模型

朱钦佩† 缪庆亮

苏州思必驰信息科技有限公司, 苏州 215000; † E-mail: ross.zhu@aispeech.com

摘要 为缓解输入语句中噪声对回复生成模型的干扰, 提出一个基于编码–解码框架的抗噪模型。首先, 在训练集输入序列中随机加入模拟噪声字符; 然后, 在编码端输出层训练噪声字符识别, 提升模型对噪声特征的提取能力; 最后, 在编码端输出层融合预训练语言模型, 扩大模型对噪声的覆盖面。为验证该模型的抗噪效果, 构建首个带真实噪声的单轮开放域闲聊系统抗噪能力测试集。在该测试集上的实验结果表明, 所提出的抗噪模型自动评测和人工评测结果均优于基准模型。

关键词 自然语言生成; 预训练语言模型; BERT; Transformer模型

An Antinoise Response Generation for Open Domain Dialogue System

ZHU Qinpei†, MIAO Qingliang

AI Speech Co., Ltd., Suzhou 215000; † E-mail: ross.zhu@aispeech.com

Abstract In order to reduce the noise interference on the response generation model, this paper proposes an antinoise model based on encoder-decoder architecture. Firstly, simulation noisy characters are added to the input utterances. Then noisy character recognition is trained at the encoder output layer, thus improving the ability of extracting noise features. Finally, pre-trained language model is fused at the encoder output layer to expand the coverage of noise. An antinoise test set is presented for verifying the model's antinoise effect, which is the first Chinese single-turn open domain dialog system corpus with real noise. Experiments show that the proposed model's results of automatic evaluation and manual evaluation on the antinoise test set are better than the baseline models.

Key words natural language generation; pre-training language models; BERT; Transformer model

近年来, 人机对话系统广泛地应用于日常生活中, 如智能导航、智能音箱和智能家居等。

在实现方法上, 目前主流的对话系统主要分为: 基于检索的对话系统和基于文本生成的对话系统。基于检索的对话系统依赖大规模的问答数据, 对数据库中存在的问句可以给出准确的回复, 但对问答库中不存在的问题无法给予有效的回复。基于生成的对话系统不依赖问答数据库, 可以根据用户输入的问题直接生成回复, 从而避免基于检索方法的缺陷。早期的文本生成方法主要基于规则生成文本, 需要人工构建大量规则, 成本较高。基于神经网络的文本生成方法直接根据训练数据, 进行端到端的学习, 不需要过多的人工干预, 受到越来越多的关注。

在交互方法上, 对话系统主要分为语音交互、文本交互和多模态交互的对话系统。语音交互对话系统因方便快捷的使用方式和无屏化操作, 广泛地应用到各个场景中。然而, 在语音交互环境中, 对话系统的输入语句往往含有大量噪声。我们从某语音对话系统中, 连续 7 天随机抽取用户请求日志 18815 条, 统计噪声语句和通顺语句的占比(表 1), 发现该系统接收的噪声语句占 34.5%。导致这种情况的主要原因是背景噪音、VAD 切分错误和 ASR 识别错误等。为分析回复生成模型的抗噪能力, 我

收稿日期: 2020─06─04; 修回日期: 2020─08─10

表 1　噪声语句和通顺语句占比

Table 1　Proportion of noisy and smooth utterances

类别	对话系统输入举例	占比/%
噪声语句	1．书山有路违纪 2．我现在有多少幽闭 3．《西游记》的读后	34.5
通顺语句	1．你的英文名字叫啥 2．你看过恐怖的东西吗 3．我可爱吗	65.5

们从该系统日志的噪声语句中随机抽取 1000 条作为测试集，实验表明，面对上述噪声输入，目前几个主流的基于神经网络的回复生成模型的回复满意率均不超过 30%，可见目前回复生成模型的抗噪能力较差。

为了提高基于生成模型的对话系统在口语环境下的抗噪能力，本文提出一个基于编码–解码框架的新模型。为验证该模型的抗噪效果，我们构建一个测试集，包含从真实对话系统脱敏日志中抽取的 1 K 噪声语句以及 10 K 人工标注无噪声问答句对。

1 相关工作

早期基于文本生成的对话系统主要通过一系列规则生成语句，这种基于规则的生成语句通常较为单一，比较生硬，人工成本高[1−3]。近年来，随着深度学习算法在自然语言处理领域的广泛应用，针对基于文本生成的对话系统的研究也逐渐转移到利用神经网络算法进行回复生成。Sutskever 等[4]提出 seq2seq，主要使用两个 LSTMs[5]结构构建编码–解码(encoder-decoder)框架，其中编码器将输入语句编码为固定维度的向量，解码器将此向量解码为可变长度的文本序列。这是一个端到端的生成模型框架，避免了人工编写规则。Bahdanau 等[6]指出，编码–解码框架中将输入语句编码为一个固定维度向量会导致信息损失，并提出注意力机制(attention mechanism)，在 seq2seq 基础上增加注意力机制(即 seq2seq+attn)，通过在解码过程中动态地“注意”输入语句的不同部分(即在生成当前时刻的词语时，只“注意”与当前时刻相关的信息)，构造出更具针对性的上下文信息进行解码。实验结果显示，注意力机制的应用大幅提升了生成语句的质量。

Vaswani 等[7]提出的 Transformer 模型放弃基于 LSTMs 的链式结构，提出自注意力机制，并将其扩展为多头自注意力机制，以此搭建整体网络。通过自注意力机制，Transformer 将任意位置的两个单词的距离转换为 1，有效地解决了自然语言处理(NLP)中棘手的长距离依赖问题，并在生成式对话任务中表现优异[8−10]。预训练语言模型可以从海量数据中进行无监督训练，学习到全面而丰富的语言学信息[11]。最近，BERT[12]在自然语言理解方面表现出强大的能力[13]。BERT 将 Transformer 作为算法的主要框架，在更大规模语料基础上，学习到更加有效的词特征表示。Sriram 等[14]在 seq2seq+attn 基础上提出 Cold Fusion 方法，在解码端中融入预训练语言模型的序列特征，提升模型的输出文本质量。

2 实现方法

首先，在输入序列中，通过随机添加和替换字符的方式，引入噪声字符。然后，在编码端的输出层识别输入序列的所有字符是否为噪声字符，并使用 Cold Fusion 融合预训练语言模型的序列特征，增强模型对输入语句噪声表示的覆盖性。最后，使用多任务训练方式，同时优化噪声预测和回复生成。为了缓解生成语句内容重复的问题，我们在解码时使用惩罚机制，降低字符再次生成的概率。模型整体框架见图 1。

2.1 抗噪机制(antinoise)

通常训练模型使用的训练集都是语句完整通顺、语义简明的句子，但这种训练集不能使模型获得处理噪声的经验，因此我们在输入语句进入模型训练之前自动添加模拟噪声，然后在编码端输出层区分噪声字符和非噪声字符。

设编码层输入序列为 EncodeInput = [w_1, w_2, …, w_N]，N 为输入序列长度。我们模拟噪声的方法是，在输入序列的 t 时刻，以概率 p_{noise} 随机增加或替换字符。经推导可知，一个输入序列不带噪声的概率为 $(1-p_{\text{noise}})^N$。假设训练过程中，不带噪声的输入语句与带噪声的语句数量比为 λ:(1–λ), $\lambda \in [0, 1]$，每个输入序列添加的噪声字符最多不超过 K 个，训练数据中所有输入语句的平均句长为 τ，则 p_{noise} 的计算

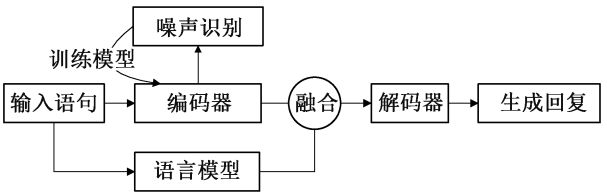

图 1　模型框架

Fig. 1　Architecture of proposed model

公式如下:

$$p_{\text{noise}} = 1 - \sqrt[\xi]{\lambda}\text{。} \tag{1}$$

增加噪声后的输入序列用式(2)表示:

$$\text{EncodeInput}_{\text{noise}} = [w_1, w_2, \ldots, w_M], \tag{2}$$

其中，M为增加噪声后的序列长度。自动标注序列中每个字符是否为噪声，噪声的训练目标用式(3)表示:

$$\text{Target}_{\text{noise}} = [\text{tgt}_1, \text{tgt}_2, ..., \text{tgt}_M], \tag{3}$$

其中，tgt_t是噪声的字符标识，$t \in [1, M]$。当输入序列中 t 时刻字符为噪声字符时，$\text{tgt}_t=1$，反之 $\text{tgt}_t=0$。

网络中对噪声的预测方法是，编码端输出的状态在每个时刻的特征向量上接入 Softmax 二分类判别器，预测当前时刻的字符是否为噪声字符。编码端输出状态为

$$\boldsymbol{S}=[\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_M], \tag{4}$$

其中，$\boldsymbol{s}_t \in \mathbb{R}^h$ 为 t 时刻字符编码特征，h 为隐藏层维度。噪声预测函数为

$$\begin{aligned}\text{Predict}_{\text{noise}} &= \text{Softmax}(\boldsymbol{W}_{\text{noise}} * \boldsymbol{S} + \boldsymbol{b}_{\text{noise}}) \\ &\triangleq [p_1, p_2, \ldots, p_M],\end{aligned} \tag{5}$$

其中，$\boldsymbol{W}_{\text{noise}} \in \mathbb{R}^{2\times h}$ 和 $\boldsymbol{b}_{\text{noise}} \in \mathbb{R}^2$ 为仿射变换参数。在计算噪声损失函数前，先对噪声优化目标 $\text{Target}_{\text{noise}}$ 做平滑归一化[15]:

$$\begin{aligned}&\text{SmoothTarget}_{\text{noise}} \\ &= [\text{smooth}(\text{tgt}_1), \text{smooth}(\text{tgt}_2), \ldots, \text{smooth}(\text{tgt}_M)] \\ &= [\text{stgt}_1, \text{stgt}_2, \ldots, \text{stgt}_M],\end{aligned} \tag{6}$$

$$\text{smooth}(\text{tgt}_t) = \begin{cases}(\varepsilon, 1-\varepsilon), & \text{tgt}_t = 0, \\ (1-\varepsilon, \varepsilon), & \text{tgt}_t = 1,\end{cases}$$

其中，$t \in [1, M]$，ε 为平滑超参数。损失函数为交叉熵与 KL 散度之和:

$$\begin{aligned}\text{NoiseLoss} = &-\sum_{t=1}^{M} \text{stgt}_t \cdot \log(p_t) - \\ &\sum_{t=1}^{M} \text{stgt}_t \cdot \log\left(\frac{\text{stgt}_t}{p_t}\right),\end{aligned} \tag{7}$$

训练过程中，噪声预测训练和回复生成训练同时进行，使用相同的学习率，使得 Trans-former 编码层除正常的学习语言特征表示外，也强化对噪声字符的表示能力。最终，输入序列在经过编码端的特征表示后参与解码，使得解码端逐渐学习到面对噪声时的回复策略，主要表现为只关注非噪声内容，或通过上下文推测噪声内容。

2.2 融合机制

为了增强网络对输入序列的编码能力，使用 Cold Fusion 融合编码端输出状态和预训练语言模型输出特征，利用预训练语言模型强大的语言表示能力，扩大模型对噪声类型的覆盖范围。假设 2.1 节中编码端输出状态为 $\boldsymbol{S}$，预训练语言模型输出序列特征为 $\boldsymbol{L}=[\boldsymbol{l}_1, \boldsymbol{l}_2, \ldots, \boldsymbol{l}_M]$，其中 $\boldsymbol{l}_t \in \mathbb{R}^d$ 为 t 时刻字符的语言模型特征，d 为语言模型输出维度。Cold Fusion 融合过程如下:

$$h_L = \text{ReLU}(\boldsymbol{W}_h \boldsymbol{L} + \boldsymbol{b}_h), \tag{8}$$

$$\boldsymbol{g}_L = \text{Sigmoid}(\boldsymbol{W}_g [\boldsymbol{S}; \boldsymbol{h}_L] + \boldsymbol{b}_g), \tag{9}$$

$$\text{output}_E = \text{ReLU}(\boldsymbol{W}_E [\boldsymbol{S}; \boldsymbol{g}_L \circ \boldsymbol{h}_L] + \boldsymbol{b}_E), \tag{10}$$

其中，$\boldsymbol{W}_h \in \mathbb{R}^{h\times d}$, $\boldsymbol{W}_g, \boldsymbol{W}_E \in \mathbb{R}^{h\times 2h}$, $\boldsymbol{b}_h, \boldsymbol{b}_g, \boldsymbol{b}_E \in \mathbb{R}^h$，都为仿射变换参数，ReLU(·)和 Sigmoid(·)为激活函数。最终使用 output_E 代替 $\boldsymbol{S}$ 作为编码端输出状态，参与解码。

2.3 抗噪+融合

将抗噪机制和融合机制联合加入编码–解码框架中，具体步骤为，在编码端输出层加入抗噪机制，并使用 Cold Fusion 融合编码端输出和语言模型输出。在结构上，抗噪机制和融合机制没有承接关系，只是共用编码器参数，各自内部参数的更新互不影响。实验中，我们在 Transformer 的基础上，同时加入抗噪机制及融合预训练语言模型 BERT，达到最优效果，二者联合的整体框架见图 2。

编码端输出与预训练语言模型输出的融合，并不影响噪声特征向解码端的传输。事实上，如果编码层的序列特征中包含噪声特征，与预训练语言模型的特征序列融合时，Cold Fusion 融合模块会自动地调节编码特征与预训练语言模型序列特征的融合比例，当编码端的噪声特征明显时，Cold Fusion 通过学习，相应地调高编码端输出的融合比例，反之亦然。

2.4 惩罚重复解码字符

生成模型的输出语句中经常出现字、词和短句重复。为了减少出现重复字词的情况，我们通过惩罚策略，降低解码字符再次出现的概率。解码端 t 时刻的输入如下:

$$\text{DecodeInput}_t = [\text{token}_1, \text{token}_2, \ldots, \text{token}_{t-1}]\text{。} \tag{11}$$

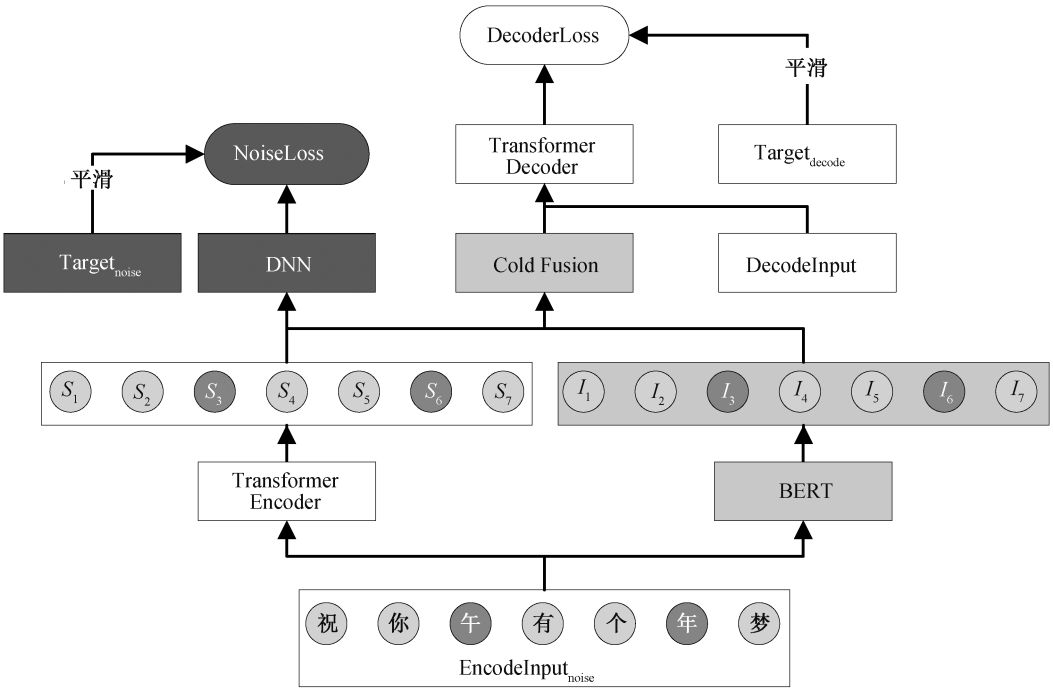

图 2　抗噪机制和融合机制联合模型

Fig. 2　Union of Antinoise and Fusion

解码端 t 时刻的输出状态为 $\boldsymbol{O}_t \in \mathbb{R}^{\text{VocabSize}}$，VocabSize 为词典大小。正常情况下，$\boldsymbol{O}_t$ 输入 Softmax 层，得到所有字典字符的概率分布 D_t，用于解码。惩罚策略的目的是降低 DecodeInput$_t$ 中字符在 D_t 上的概率。惩罚过程为

$$\text{Mask}_t = \text{OneHot}(\text{DecodeInput}_t, \text{VocabSize}), \quad (12)$$

$$\text{Penalty}_t = \left|\boldsymbol{O}_t \circ \text{Mask}_t\right| \cdot \delta, \quad (13)$$

$$\widehat{\boldsymbol{O}}_t = \boldsymbol{O}_t - \text{Penalty}_t, \quad (14)$$

$$\widehat{D}_t = \text{Softmax}(\widehat{\boldsymbol{O}}_t), \quad (15)$$

其中，OneHot 为独热编码，编码维度为 VocabSize。δ>0 为超参数。最终，$\widehat{D}_t$ 代替 D_t 参与 t 时刻解码。

惩罚重复字符的方法不参与模型训练，直接在训练完成的模型上使用，可在一定程度上避免生成无意义重复词句的情况。

3　实验

3.1　训练数据和测试数据

训练数据有 4 个来源：1) 人工标注闲聊问答语句对 740 K，即根据用户请求语句，标注人员以智能机器人的口吻写出通顺、合理和有趣的回复；2) DuReader[16]语料 300 K，只使用语料中的问答句对；3) Weibo[17]语料 2 M，只使用所有对话中的首轮对话；4) 收集脱敏日志噪声请求语句 10 K，目标回复语句均设置为“[SafeRes]”，表示这些回复语句应使用安全回复，比如“抱歉，这个我还不会”。所有语料问句和回复语句长度均在 2~30 之间，去除问句中的标点符号，但保留回复语句中的标点符号。将所有数据汇总，随机抽取 100 K 作为验证集，其余用于训练。

在验证一个回复生成模型抗噪能力的同时，要求该模型在通顺语句中的效果不下降。我们构建的测试集也主要针对这两个方面。测试集以及评测结果可以从 GitHub①获取，测试集举例见表 2。

表 2　测试集

Table 2　Test data set

名称	举例	数量
QA10K	Q1: 想问你最喜欢的人是谁 A1: 我不说，相信时间会给你答案的 Q2: 祝你有个好梦 A2: 梦里有你就是好梦呢	10000
Hard1K	1. 我现在有多少幽闭 2. 书山有路违纪	1000

① https://github.com/zqp2009happy/Antinoise_FuseBERT.git

QA10K 随机抽取 10 K 对人工标注闲聊问答句对。这个测试集的所有请求语句和人工标注回复语句均表达清晰, 语句通顺。评测指标为 BLEU[18]、distinct-N[19]和平均回复句长。

Hard1K 从日志分析的噪声语句中, 随机抽取 1000 条, 主要测试模型对噪声语句的处理能力。由于人工很难根据此测试集的噪声语句标注回复, 更具操作性的评测方法是, 当模型生成回复后, 人工评测回复语句的合理性, 评测指标为人工评测、distinct-N 和平均回复句长。

利用项目 sacrebleu①计算 BLEU 值, 利用 GitHub②计算 Distinct-1 和 Dinstinct-2。平均回复句长由下式计算:

$$\text{平均回复句长}=\frac{\text{所有回复语句字数总和}}{\text{回复语句个数}}。\quad(16)$$

3 位数据标注人员对所有模型的回复语句进行独立标注, 标注过程中屏蔽回复语句和模型的对应关系。每个回复语句标注满意或非满意: 1) 满意, 表示回复语句通顺易懂, 并且输入语句和回复语句有合理的承接和逻辑关系; 2) 非满意, 表示回复语句语句不通, 语义不明, 或者“答非所问”。每个回复语句的最终标注结果采用少数服从多数的原则。人工评测结果由下式计算:

$$\text{人工评测}=\frac{\text{标注“满意”句子个数}}{\text{所有句子个数}}。\quad(17)$$

3.2 模型评测

为验证各个机制的性能以及多机制融合方法的有效性, 设置 3 组对比实验, 分别是基础模型加抗噪机制, 基础模型加语言模型融合机制, 基础模型同时加抗噪机制和融合机制。3 组对比实验分别在 QA10K 和 Hard1K 上进行。基线模型选择 seq2seq+attn 和 Transformer, 具体配置如下。

seq2seq+attn 模型工程代码引用项目 TEXAR③。隐藏层大小(hidden_size)设置为 512, 定向搜索(beam search)宽度为 10。

Transformer 模型工程代码引用项目 TEXAR④。隐藏层大小设置为 512, 编码和解码层数(block_num)设置为 6, 注意力机制头个数为 8, 定向搜索宽度为 10。

InputNoise 在 Transformer 的基础上, 只在训练集的输入语句加入噪声, 但不训练噪声。

Antinoise 在 Transformer 的基础上, 加入抗噪机制。训练数据输入语句的平均句长 τ=12, 令 λ=0.5, 则噪声概率 p_{noise}=0.056, 最大噪声数量 K 为当前句长的 1/2, 平滑参数 ε=0.9。

FuseBERT 使用 Cold Fusion 融合 Transformer 编码输出与 BERT 输出, 不加抗噪机制。BERT 工程代码引用 Google-Research 官方代码⑤, 预训练参数使用 110 M 中文模型(BERT-Base, Chinese)。使用训练集所有输入语句对 BERT 做预训练, 在 BERT-Base 的基础上微调 925000 步, 掩码预测精度为 0.72。在训练过程中, 更新 BERT 参数。

Antinoise+FuseBERT Transformer 编码端输出层接入抗噪机制, BERT 不参与噪声学习, 用 Cold Fusion 融合 Transformer 编码输出和 BERT 输出, 沿用上述各模型参数的设置。

上述所有模型均不分词, 使用 BERT 自带词典。模型批训练数据大小(batch_size)为 2048, 解码目标语句标签平滑归一化参数为 0.9, 最大解码长度为 32。惩罚重复字词超参数 δ=2.0, 最大 Epoch 为 20, 保存验证集上效果最优模型, 用于实验比对。所有模型都可能生成特殊回复语句“[SafeRes]”, 表示本句采用安全回复, 此特殊回复语句的句长为 1, Distinct-1 得分默认为 0, 人工评测默认不满意。3 位标注人员人工评测的 Kappa 系数一致率平均值为 0.82, 表明评测结果具有高度一致性。实验结果见表 3。

由表 3 可以看出, 我们的模型 Antinoise, FuseBERT 和 Antinoise+FuseBERT, 在各评测指标中超过基线模型 seq2seq+attn 和 Transformer, 且联合模型 Antinoise+FuseBERT 效果最优, 说明我们的模型不仅具有一定的抗噪能力, 对常规通顺语句也保持良好的效果。在 QA10K 和 Hard1K 上, InputNoise 各个指标几乎均低于基线模型 Transformer, 说明仅仅模拟噪声输入, 并不能有效地提取噪声特征, 反而使模型在训练中陷入困惑。除此之外, 基线模型 seq2seq+attn 各个指标均低于 Transformer 模型。

① https://pypi.org/project/sacrebleu/1.1.7; ② https://github.com/neural-dialogue-metrics/Distinct-N; ③ https://github.com/asyml/texar/example/seq2seq_attn;④ https://github.com/asyml/texar/example/transformer; ⑤ https://github.com/google-research/bert

表 3　QA10K 和 Hard1K 的测试结果

Table 3　Test results on QA10K and Hard1K

模型	QA10K				Hard1K			
	BLEU	Distinct-1	Distinct-2	平均回复句长	人工评测/%	Distinct-1	Distinct-2	平均回复句长
seq2seq+attn	4.3	0.678	0.678	9.69	10.1	0.591	0.587	8.52
Transformer	6.1	0.954	0.851	10.3	25.4	0.698	0.599	6.56
InputNoise	5.2	0.859	0.775	8.58	18.0	0.719	0.631	6.53
Antinoise	6.7	0.966	0.871	10.9	33.2	0.876	0.764	8.83
FuseBERT	6.2	0.956	0.857	10.7	32.1	0.731	0.632	7.41
Antinoise+FuseBERT	**6.7**	**0.978**	**0.884**	**11.5**	**40.5**	**0.911**	**0.806**	**9.92**

说明：粗体数字表示效果最佳，下同。

对 Hard1K 回复内容进一步分析可以发现，加入抗噪机制的回复语句使得模型更加关注输入语句的有效部分，倾向于忽略“不理解”的语义内容，从而采取比较安全的方式回复用户请求，这是人工评测指标大幅度提升的关键，也是 Antinoise 效果超过 FuseBERT 的主要原因，符合我们模型设计的预期。比如，输入语句“我现在有多少幽闭”，其中“幽闭”在这里明显不符合正常说法，Transformer 模型生成安全回复标志语句“[SafeRes]”；FuseBERT 模型缺少对噪声和非噪声字符的区分，总是尝试理解输入语句的所有内容，它结合对“幽闭”理解和输入语句其他部分，回复为“哈哈，你想幽闭的话可以自己数一数呀”，导致回复内容不符合正常说法；Antinoise 模型对“幽闭”这个词视而不见，回复为“你自己数一数就知道了呀”；Antinoise+FuseBERT 同样忽略“幽闭”，回复为“你看一下就知道啦”。可见加入抗噪机制后，回复内容并没有提及“幽闭”相关内容，似乎模型知道用户问某样东西有多少，但模型并不关心是什么东西，从而生成较为通用的回复，从用户角度看，这种回复并非不妥。

另一方面，FuseBERT 对模型语义理解能力的提升也不容忽视。比如，输入语句“书山有路违纪”，Transformer 模型不理解输入语义，回复内容照搬输入“书山有路违纪”；Antinoise 模型知道输入语句中包含噪声，但知识覆盖面不足，回复为“是的呢，这句诗出自李清照《水浒传》哦”；FuseBERT 和 Antinoise+FuseBERT 由于语言模型 BERT 的存在，从而抓住“书山有路”这个有效且强烈信息，都回复为“书山有路勤为径，学海无涯苦作舟”。更多的 Hard1K 回复举例见表 4。

4 结语

在语音交互环境下，输入对话系统带噪声的语句占整体输入的 34.5%，但主流的回复生成模型抗噪能力不佳。为了提高回复生成模型的抗噪能力，本文基于编码–解码结构，构建一个新网络，极大地提升了编码端对噪声特征的表示能力。同时，本文公开首个中文单轮开放域闲聊系统的抗噪能力测试集，包含 10 K 人工标注问答对以及 1 K 噪声语句，作为带真实噪声的对话数据，有助于其他研究者针

表 4　Hard1K 测试回复举例

Table 4　Response examples on Hard1K

输入语句	Transformer	Antinoise	FuseBERT	Antinoise+FuseBERT
一年有	是啊，一年	一年有什么	[SafeRes]	**一年有三百六十五天哦。**
书山有路违纪	书山有路违纪	是的呢，这句诗出自李清照《水浒传》哦	**书山有路勤为径，学海无涯苦作舟。**	**书山有路勤为径，学海无涯苦作舟。**
我现在有多少幽闭	[SafeRes]	**你自己数一数就知道了呀**	哈哈，你想幽闭的话可以自己数一数呀	**你看一下就知道啦**
我是要乘	好吧，那我就不说话了	**你要乘什么**	我是你的智能助手，很高兴认识你。	**你要乘什么**
讲个小妖	[SafeRes]	讲个故事哄我开心吧	小可爱想听什么都可以哦	**你想让我讲什么，麻烦说清楚一点哦**

对汉语语音对话系统的相关研究。在该测试集上的实验结果表明, 抗噪机制和融合BERT方法均优于基准模型, 并且二者联合模型达到最优效果。

在未来工作中, 我们将尝试使用更加逼真的噪声模拟策略, 比如根据语音识别中的语言模型模拟噪声, 同时使用增加、替换、删除和重复等策略模拟噪声。同时, 将尝试在模型不同位置上训练噪声, 比如解码输出状态与BERT融合后, 再训练噪声。此外, 还将寻找或研究更加合理的自动评测指标。

参考文献

[1] Collby K M. Artificial paranoia: a computer simulation of paranoid process. New York: Elsevier Science Inc, 1975

[2] Barzilay R, Lee L. Catching the drift: probabilistic content models, with applications to generation and summarization // Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. Boston, 2004: 113–120

[3] Angeli G, Liang P, Klein D. A simple domain-independent probabilistic approach to generation // Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Cambridge, 2010: 502–512

[4] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks // Advances in Neural Information Processing Systems. Montreal: MIT Press, 2014: 3104–3112.

[5] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997, 9(8): 1735–1780

[6] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [EB/OL]. (2016–05–19) [2020–06–01]. https://arxiv.org/abs/1409.0473

[7] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. Advances in Neural Information Processing Systems, 2017: 5998–6008

[8] Wolf T, Sanh V, Chaumond J, et al. Transfertransfo: a transfer learning approach for neural network based conversational agents [EB/OL]. (2019–02–04) [2020–06–01]. https://arxiv.org/abs/1901.08149

[9] Rashkin H, Smith E M, Li M, et al. Towards empathetic open-domain conversation models: a new benchmark and dataset // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, 2019: 5370–5381

[10] Dinan E, Roller S, Shuster K, et al. Wizard of Wikipedia: knowledge-powered conversational agents [EB/OL]. (2019–02–21) [2020–06–01]. https://arxiv.org/abs/1811.01241

[11] Jozefowicz R, Vinyals O, Schuster M, et al. Exploring the limits of language modeling [EB/OL]. (2016–02–11) [2020–06–01]. https://arxiv.org/abs/1602.02410

[12] Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding // Proceedings of NAACL-HLT. Minneapolis, 2019: 4171–4186

[13] Zhu J, Xia Y, Wu L, et al. Incorporating bert into neural machine translation [EB/OL]. (2020–02–17) [2020–06–01]. https://arxiv.org/abs/2002.06823

[14] Sriram A, Jun H, Satheesh S, et al. Cold fusion: training Seq2Seq models together with language models [EB/OL]. (2017–08–21) [2020–06–01]. https://arxiv.org/abs/1708.06426

[15] Szegedy C, Vanhoucke V, Loffe S, et al. Rethinking the inception architecture for computer vision // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016: 2818–2826

[16] He W, Liu K, Liu J, et al. DuReader: a Chinese machine reading comprehension dataset from real-world applications // Proceedings of the Workshop on Machine Reading for Question Answering. Melbourne, 2018: 37–46

[17] Zheng Y, Chen G, Huang M, et al. Personalized dialogue generation with diversified traits [EB/OL]. (2020–01–02) [2020–06–01]. https://arxiv.org/abs/1901.09672

[18] Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. Philadelphia, 2002: 311–318

[19] Li J, Galley M, Brockett C, et al. A diversity-promoting objective function for neural conversation models // Proceeding of NAACL-HLT. San Diego, 2016: 110–119