ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 31,No. 2,Mar. 2016,pp. 347－354
DOI:10.16337/j.1004-9037.2016.02.015

http://sjcj.nuaa.edu.cn
E-mail:sjcj@nuaa.edu.cn
Tel/Fax:＋86-025-84892742

基于循环神经网络语言模型的 N-best 重打分算法

张 剑 屈 丹 李 真

(解放军信息工程大学信息系统工程学院,郑州,450002)

摘 要:循环神经网络语言模型能够克服统计语言模型中存在的数据稀疏问题,同时具有更强的长距离约束能力,是一种重要的语言模型建模方法。但在语音解码时,由于该模型使词图的扩展次数过多,造成搜索空间过大而难以使用。本文提出了一种基于循环神经网络语言模型的 N-best 重打分算法,利用 N-best 引入循环神经网络语言模型概率得分,对识别结果进行重排序,并引入缓存模型对解码过程进行优化,得到最优的识别结果。实验结果表明,本文方法能够有效降低语音识别系统的词错误率。
关键词:语音识别;语言模型;循环神经网络;N-best 重打分;缓存语言模型
中图分类号:TP391　　**文献标志码**:A

N-best Rescoring Algorithm Based on Recurrent Neural Network Language Model

Zhang Jian, Qu Dan, Li Zhen

(Institute of Information Systems Engineering, PLA Information Engineering University, Zhengzhou, 450002, China)

Abstract: Recurrent neural network language model (RNNLM) is an important method in statistical language models because it can tackle the data sparseness problem and contain a longer distance constraints. However, it lacks practicability because the lattice has to expand too many times and explode the search space. Therefore, a N-best rescoring algorithm is proposed which uses the RNNLM to rerank the recognition results and optimize the decoding process. Experimental results show that the proposed method can effectively reduce the word error rate of the speech recognition system.
Key words: speech recognition; language model; recurrent neural network; N-best rescoring; cache language model

引　　言

语音是人类进行思想、观点和情感交流最有效、最便捷的方式之一,同时也是重要的人机交互手段[1]。语音识别的研究目标就是让机器“听懂”人类的语言,它是一种模拟人类认知语言的计算过程,将数字语音信号识别成语言符号序列,从而达到对语音信号的理解。而如何让计算机有效地理解语音中承载的自然语言信息,能够“听懂”人类的语言,很大程度上需要依赖语言模型技术的不断发展。

基金项目:国家自然科学基金(61175017)资助项目;国家高技术研究发展计划(“八六三”计划)(2012AA011603)资助项目;全军军事学研究生课题(2010JY0258-144)资助项目。
收稿日期:2013-11-20;**修订日期**:2014-01-06

语言模型(Language model, LM)是用来描述自然语言内在规律的数学模型,其需要发现、归纳和获取自然语言在统计和结构方面的内在规律,从而成为计算机可处理的语言。语言模型已成为自动语音识别、机器翻译和信息检索等系统的重要组成部分。语言模型通常分为基于规则的语言模型和统计语言模型[2]。目前统计语言模型在实际应用中处于主流地位,它通过概率和分布函数来描述词、词组及句子等自然语言基本单位的性质和关系,体现了自然语言中存在的基于统计原理的生成和处理规则[3]。其中,N 元文法(n-gram)语言模型是应用最广泛的一种,但其仍存在很多的不足,主要表现为:首先由于语言模型建模受到训练语料规模的限制,其分布存在一定片面性,一些合理的语言搭配没有出现在训练文本中,导致数据稀疏问题[4],同时由于马尔科夫假设限制 N 的大小,只能对短距离的词之间的转移关系进行建模,无法体现长距离的词之间的依赖关系;其次现有语言模型大部分只用到字、词等语法层面的简单信息,很少使用到深层的语言知识,描述能力较差,不能很好地反映真实的概率分布[5]。

因此,近年来学者们在统计语言模型领域做了很多的研究,提出了很多不同的方法来改善语言模型的性能,其中神经网络语言模型(Neural network language model, NNLM)由于具有较好的性能,是目前研究的热点。相比于传统的稀疏表示法,神经网络语言模型采用低维实数向量表示词,在连续空间中计算语言模型概率,成功克服了数据稀疏的影响,其中最典型的算法是文献[6]提出的前馈神经网络语言模型,目前在语音识别、机器翻译等领域中得到了很好的应用。

文献[7]发现,前馈神经网络语言模型获得长距离信息的能力仍有不足,因而提出循环神经网络语言模型(Recurrent neural network language model, RNNLM),相关实验表明,其模型的性能表现更佳。但同时由于循环神经网络语言模型需要预测词的全部上下文信息,在解码的过程中造成 Lattice 扩展次数过多,搜索空间巨大,很难进行 Lattice 重打分。

为了克服上述问题,本文充分利用循环神经网络语言模型和高阶 n-gram 语言模型的优势,采用混合模型对 N-best 进行二次重打分,避免了 Lattice 搜索空间巨大的问题,并且在解码时引入缓存模型,解决了测试数据和训练数据不匹配的问题。实验结果表明,这种方法能够有效降低语言模型的困惑度,同时对系统的识别正确率也有一定的提高。

1 循环神经网络语言模型

1.1 循环神经网络语言模型网络结构

常见的循环神经网络结构有 Elman 网络、Jordan 网络和 Hopfield 网络等,相比于前馈神经网络具有更好的训练能力。文献[7]提出的循环神经网络语言模型采用 Elman 网络,相比于前馈神经网络语言模型,其主要在于词的历史信息的表示方法不同,前馈神经网络模型与 N 元文法模型相似,仍采用前 $N-1$ 个词表示其历史信息,而循环神经网络模型则通过训练过程中对数据的学习得到历史信息,在理论上可以表示更长的上下文历史信息,此外这种表示方法也可以包含更多高层次的语言知识,如句法、语义等[8]。其循环神经网络结构如图 1 所示。

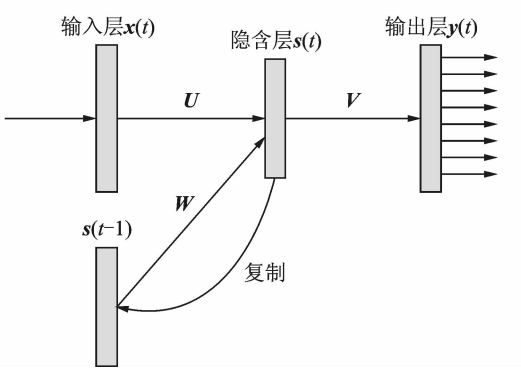

图 1 循环神经网络结构

Fig. 1 Architecture of recurrent neural network language model

其网络结构包括输入层、隐含层和输出层。在 t 时刻,网络的输入为 $\boldsymbol{x}(t)$,输出为 $\boldsymbol{y}(t)$,隐含层 $\boldsymbol{s}(t)$ 表示网络的状态。输入向量 $\boldsymbol{x}(t)$ 由当前词向

量 $\boldsymbol{w}(t)$(采用 One-hot 表示,向量大小与词汇表大小 V 相同)和上一时刻隐含层的输出 $\boldsymbol{s}(t-1)$组成。网络训练完成后,输出向量 $\boldsymbol{y}(t)$表示待预测词 w_{t+1}在给定当前词 w_t 和历史信息 $\boldsymbol{s}(t-1)$时的概率 $P(w_{t+1}|w_t,\boldsymbol{s}(t-1))$。网络中,输入层没有计算能力,按训练语料的表示方法对语料进行预处理;隐含层中处理输入神经元传递进来的信息;输出层则表示网络的输出结果。层与层之间的神经元靠突触传递信息,网络中各层之间的连接用权值矩阵表示,其中,$\boldsymbol{U}$,$\boldsymbol{W}$ 为输入层与隐含层之间的权值矩阵,$\boldsymbol{V}$ 为隐含层与输出层之间的权值矩阵,则网络中各层输出值用矩阵形式表示为

$$\boldsymbol{x}(t)=\boldsymbol{w}(t)+\boldsymbol{s}(t-1) \tag{1}$$

$$\boldsymbol{s}(t)=f(\boldsymbol{U}\boldsymbol{w}(t)+\boldsymbol{W}\boldsymbol{s}(t-1)) \tag{2}$$

$$\boldsymbol{y}(t)=g(\boldsymbol{V}\boldsymbol{s}(t)) \tag{3}$$

式中:$f(z)$为 sigmoid 激活函数 $f(z)=\dfrac{1}{1+\mathrm{e}^{-z}}$;$g(z)$为 softmax 激活函数 $g(z_m)=\dfrac{\mathrm{e}^{z_m}}{\sum_k \mathrm{e}^{z_m}}$。

网络的输出 $\boldsymbol{y}(t)$表示待预测词 $\boldsymbol{w}(t+1)$在给定当前词 $\boldsymbol{w}(t)$和历史信息 $\boldsymbol{s}(t-1)$条件下的概率分布。其训练和测试时每步的时间复杂度正比于

$$O=H\times H+H\times V=H\times(H+V) \tag{4}$$

式中:H 为隐含层大小;V 为词汇表大小。

1.2　基于词类的循环神经网络语言模型

神经网络语言模型过高的计算复杂度为其在识别系统中的应用造成了极大的限制,成为其发展的瓶颈。其中,隐含层和输出层之间存在的高维矩阵运算是主要的问题所在。例如,在大词汇量语音识别系统中,当采用 $H=200$, $V=50\ 000$ 时,模型的训练时间过长会使模型在识别系统中的可用性过差。

通常,由于 $H\ll V$,因此这方面的工作主要集中于输出层结构的改进,降低输出词汇表 V 的大小。例如文献[9]提出用 Short-list 列表来限制输出层大小(列表大小 $s\ll V$),对其中的 s 个高频词用神经网络语言模型计算其概率,其他的低频词则通过 n-gram 模型得到[9]。这种方法在一定程度上能够提高训练速度,但当 s 值较小时,会使模型性能明显下降[10]。而文献[11]引入类别层的概念,提出基于类的 RNNLM(Class based RNNLM, CRNNLM),根据词频对词汇表进行分组,将输出层分解。其假设每个词都有唯一对应的词类,在输出层计算时,先利用 RNN 计算词类的概率分布,再从所需的词类中计算特定词的概率。因此,式(3)中的输出层概率计算变为

$$\boldsymbol{c}(t)=g(\boldsymbol{X}\boldsymbol{s}(t)) \tag{5}$$

$$\boldsymbol{y}(t)=g(\boldsymbol{V}'\boldsymbol{s}(t)) \tag{6}$$

式中:$\boldsymbol{V}'$为 $\boldsymbol{V}$ 的子集,为各词类对应的权值矩阵;$\boldsymbol{X}$ 为类别层和隐含层之间的权值矩阵,则词 w_{t+1}的概率由可计算为

$$P(w_{t+1}\mid\boldsymbol{s}(t))=P(c_i\mid\boldsymbol{s}(t))P(w_i\mid c_i,\boldsymbol{s}(t)) \tag{7}$$

式中:w_i 为预测词 w_{t+1}的索引;c_i 为其所属词类。改进后的模型时间复杂度正比于

$$O=H\times H+H\times C=H\times(H+C) \tag{8}$$

式中:C 为类别层的个数。由于所需词类的个数远小于词汇表的大小,所以改进的 CRNNLM 能够有效降低模型的计算复杂度。在海量数据的情况下,输出层往往十分巨大(超过 100 000),采用 CRNNLM 对模型训练速度的提升更为明显。

2　循环神经网络模型的重打分算法

2.1　连续语音识别系统

连续语音识别系统一般可以分为训练和识别两个阶段,主要包含特征提取、声学模型、语言模型和解码网络 4 部分[12]。首先对语音数据进行前端处理,提取语音信号的特征参数,将语音信号转化为对

应语音帧的高维特征矢量。训练过程中，利用训练语料得到声学模型、语言模型和词汇树等。解码器将训练阶段和识别阶段联系起来，基于最大后验概率准则进行解码，其输出是以图或网格表示的多候选结果，最后对解码结果进行后处理(二次解码)得到最终的识别结果。其系统框架如图 2 所示。在一次解码中，出于解码速度的考虑，采用低阶的 3-gram 模型，得到中间识别结果，在二次解码中利用 RNNLM 和高阶 4-gram 对中间结果进行重打分，提高解码精度，得到最优的识别结果。

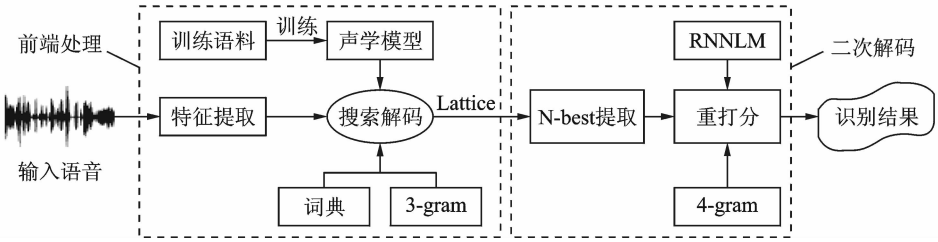

图 2 连续语音识别系统

Fig. 2 Structure of continuous speech recognition system

2.2 RNN 重打分算法

语音识别的解码过程，就是解码器利用先验知识(声学模型、语音模型和词典等)，从状态空间中找到最优的状态序列，本质是一个搜索问题。目前常用的解码策略按先验知识的利用方式，可分为一遍解码策略和多遍解码策略。

在一遍解码中使用的知识源越多，使搜索空间复杂性过大，搜索的代价就越大，因此很难应用于比较复杂的声学模型和语言模型。目前很多语音识别系统都采用多遍解码的策略，引入各种复杂的声学模型和语言模型，提高系统识别的正确率。

由于解码过程都基于最大后验概率准则，这样得到的识别结果是整体句子的最大后验概率得分。而语音识别中通常采用词错误率作为评价标准，这与最小化句子错误率的解码准则不相匹配。一般采用多遍解码策略，通过二次解码来实现识别结果的最小化词错误率解码。常用的实现方式有基于 N-best 的方法和基于 Lattice 的方法。

Lattice 作为一个有向无环图，相比于 N-best 的线性结构，包含更多的假设空间。由于循环神经网络语言模型采用全部的上下文信息来预测下一个词，对 Lattice 重打分时需要更多的 Lattice 扩展次数，造成搜索空间的过大，不适宜引入 RNN 模型进行重打分。而 N-best 结构则更适合这种包含长距离信息的模型，因此采用 N-best 进行 RNN 重打分[13]。其基本思想是在每个时间点保存多条路径，当两个或更多路径同时到达一个相同的状态时，拥有相同词历史的路径将被合并，新路径的概率是原来多个路径的概率和；同一时刻每个状态将最多保存 N 条不同词历史的最佳路径。在解码结束时，将最终状态的 N 个路径经过简单的重新排序就可以得到 N 个最佳词序列。

在一次解码生成的 Lattice 中，提取 N-best 列表，得到最佳的 N 条假设，表示为 $\pi_1, \pi_2, \cdots, \pi_N$。路径的排序方式按照声学模型和语言模型的联合似然概率得分从大到小排列，如

$$A[\pi_1]L[\pi_1]^{\gamma} \geqslant A[\pi_2]L[\pi_2]^{\gamma} \geqslant \cdots \geqslant A[\pi_N]L[\pi_N]^{\gamma} \tag{9}$$

式中：γ 为语言模型比例参数；$A[\pi]$和 $L[\pi]$分别为声学模型得分和语言模型得分。

在二次解码时，引入新的语言模型进行重打分，更新每个 N-best 假设的对数似然得分，得到最优路径。其路径的对数似然概率得分为

$$\log L(s) = n \cdot wp + \sum_{i=1}^{n} asc_i + \mathrm{lms} \sum_{i=1}^{n} \log P_x(w_i \mid h_i) \tag{10}$$

式中：n 为词个数；wp 为词的插入代价；asc_i 为词 w_i 的声学模型得分；h 为词的历史 $w_1, \cdots, w_{i-1}$；lms 为

语言模型尺度因子;P_x 为语言模型得分。

采用循环神经网络语言模型与高阶 n-gram 进行语言模型重打分,更新后的语言模型得分为

$$P_x(w_i \mid h)=\lambda P_{\text{rnn}}(w_i \mid h)+(1-\lambda)P_{n\text{-gram}}(w_i \mid h) \tag{11}$$

式中:λ 为 RNNLM 的权重;P_{rnn} 和 $P_{n\text{-gram}}$ 分别为循环神经网络模型和 n-gram 模型概率。

2.3　缓存循环神经网络模型

在统计语言模型中,训练数据和测试数据的不匹配是影响模型识别性能的一个重要因素。大多数情况下,训练数据和测试数据中的词汇并不完全相同,语言模型在测试集上进行识别时,经常会遇到训练过程中并未出现的词,如人名等,仅靠原有的模型不能准确地计算出其概率。

在传统的统计语言模型中,常采用 Cache 模型来处理这一问题。这是一种动态自适应的方法,在识别过程中,将系统缓存区的数据作为一个动态模块,作为词的历史信息来训练缓存模型,并与原有 n-gram模型通过插值法结合,可表示为

$$P(w_i \mid h)=\lambda P_{\text{Cache}}(w_i \mid h)+(1-\lambda)P_{n\text{-gram}}(w_i \mid h) \tag{12}$$

式中:$p_{\text{Cache}}(w_i \mid h)$为缓存模型的概率。

本文引入缓存神经网络模型,在神经网络对测试数据进行识别的同时,对其进行学习,更新各层神经元的权值,并计算得到词序列的缓存模型的概率得分。这样在对 N-best 路径进行重打分时,将不同的模型得分相结合,最终的语言模型得分为

$$P(w_i \mid h)=\lambda P_{\text{rnn}}(w_i \mid h)+\mu P_{\text{cache-rnn}}(w_i \mid h)+(1-\lambda-\mu)P_{n\text{-gram}}(w_i \mid h) \tag{13}$$

式中:$P_{\text{cache-rnn}}(w_i \mid h)$为缓存神经网络模型的概率。

3　实验结果与分析

3.1　实验配置

本文的实验数据采用微软语料库 Speech Corpora (Version 1.0),其中训练集包含 100 个年龄在 18～40 岁间的男性录音,每人大约 200 句,总共 19 688 句,454 315 个音节,约 33 h 的语音数据;测试集包含另外 25 个男性录音,每人 20 句,共 500 句话。语音采样频率为 16 kHz,采用 16 bit 线性 PCM 量化。

本文采用 Kaldi 工具箱[14]搭建连续语音识别系统。语音的声学特征采用 13 维梅尔频率例谱系数(Mel-frequency cepstral coefficients,MFCC)参数及其一阶、二阶差分系数,共 39 维特征矢量。语音信号采用 Hamming 窗处理,帧长 25 ms,帧移 10 ms。声学模型采用有调音节的声韵模型,共 187 个模型基元,采用三状态自左向右无跨越的隐马尔科夫模型(Hidden Markov model, HMM)。此外,由于语音中存在协同发音现象,因而在声韵母结构上,采用三音子的声韵母结构。语言模型的训练数据采用微软语料库中的语音标注文件,共含 454 315 个音节,19 688 句话,采用 SRILM 工具包训练得到三元文法和四元文法模型,采用 Kneser-Ney 平滑技术。

3.2　语言模型评测标准

在评价语言模型时,常采用信息论中的交叉熵和困惑度(Perplexity, PPL)的方法来评测模型性能。

测试语言模型的性能时,常需要求模型在给定测试集上的交叉熵。交叉熵的概念是用来衡量语言模型和自然语言中的真实概率分布之间的差异情况。其反映了每个词平均携带的信息量,交叉熵越小,模型更接近真实概率分布,性能也就越好。

对于语言模型 M 给定的测试集 T 由句子序列 $t_1,t_2,\cdots,t_{l_T}$ 组成,则模型在测试文本上的交叉熵定义为

$$H(P_T,P_M)=-\frac{1}{N_T}\sum_{i=1}^{l_T}\log_2 p_M(t_i) \tag{14}$$

式中：N_T 为当前测试集 T 中词的总个数；l_T 为测试集 T 中包含的句子个数；t_i 为 T 中一个句子，$p_M(t_i)$ 为句子 t_i 由模型 M 计算得到的概率。

在交叉熵的基础上，可进一步定义语言模型的困惑度，这也是评测语言模型性能时最常用的指标。困惑度为

$$\mathrm{PPL}=2^{H(P_T,P_M)}=2^{-\frac{1}{N_T}\sum_{i=1}^{l_T}\log_2 p_M(t_i)}=\left[\prod_{i=1}^{l_T}p_M(t_i)\right]^{-\frac{1}{N_T}} \tag{15}$$

困惑度可认为是语言模型在预测某种语言现象时每个词后的候选词的几何平均数。困惑度越低，语言模型对上下文的约束能力越强，说明其对语言的表述能力越强，因而具有更好的模型性能。

3.3 实验结果及分析

3.3.1 循环神经网络语言模型性能

在神经网络模型中，隐含层神经元个数是影响网络性能的最重要的因素。若神经元个数较少，则网络不能充分描述输出与输入变量之间的关系；相反，若神经元个数过多，则会导致网络的学习时间变长，甚至出现过拟合的问题。因此，为观察隐含层大小对模型性能的影响，设置不同的神经元个数，训练多组 RNN 模型。通常神经网络中隐含层神经元个数为 30～600，根据语料库大小的不同，所需要的神经元个数也略有不同。

本文实验的训练数据采用微软语料库中的训练集语音标注文件，测试数据为测试集语音标注文件。根据不同模型参数，训练 RNN 模型，在测试数据上计算模型的困惑度，并与 N 元文法模型进行对比。

表 1 给出了不同隐含层大小下的 RNN 语言模型模型与采用 Kneser-Ney 平滑的 4-gram 模型(KN4)的对比。本文选取的神经元个数为 50～500，RNN-N 表示隐含层神经元个数为 N。

表 1 RNN 模型与 4-gram 模型的性能比较

Tab. 1 Performance comparison between RNNLM and 4-gram model

模型	困惑度	
	RNN	RNN+KN4
KN-4-gram		99.3
RNN-50	136.0	97.5
RNN-100	129.1	94.5
RNN-150	123.0	93.9
RNN-200	121.6	93.9
RNN-250	122.1	94.0
RNN-300	122.4	93.9
RNN-350	123.8	94.1
RNN-400	123.2	94.3
RNN-450	123.2	94.3
RNN-500	125.6	95.1

从表 1 中可看出，不同隐含层大小训练得到的 RNN 模型，均能有效降低 n-gram 模型的困惑度，证明此方法能够提高语言模型的性能。随着神经元个数的增加，RNN 模型的性能逐渐提高，当神经元个数为 200 时，达到最好的模型性能，神经元个数再次增加，模型性能变差，表明网络中出现过拟合现象。

另一方面，神经网络语言模型的训练往往需要消耗大量的时间，很大程度上限制了模型的应用。引入类别层对输出层结构进行改进，通过降低隐含层和输出层之间矩阵计算的维数，来改善训练时间过长的情况。因此，本实验选取不同大小的类别层，在相同参数条件下训练 RNN 模型，比较模型的训练时间和在测试数据上的困惑度，实验结果如表 2 所示。其中，隐含层神经元个数为 300，采用通过时间的反向传播(Back-propagation through time, BPTT)算法迭代 5 次，FULL 表示不采用类别层的 RNN 模型。

表 2　类别层个数对 RNN 模型性能影响及所需训练时间

Tab. 2　Performance of RNNLM with class layer and training time

类别层	RNN	RNN+KN4	训练时间	识别时间/s
40	122.5	94.4	33 min 5 s	1.264
80	123.3	94.8	28 min 19 s	1.404
120	123.1	93.8	35 min 21 s	1.756
160	121.3	93.9	27 min 21 s	1.513
200	122.5	93.8	47 min 27 s	1.591
240	120.5	93.4	44 min 26 s	1.638
280	120.4	93.3	44 min 11 s	1.778
320	121.7	93.8	34 min 46 s	1.888
800	119.0	91.9	64 min 6 s	3.182
FULL	116.0	91.4	172 min 12 s	4.415

实验结果表明,当不改变输出层结构时,原有 RNN 模型的训练需要大量的时间,接近 3 h,代价过高,给实际应用造成很大的不便。而对比 CRNNLM 的训练时间,可以看到,通过类别层的引入,使模型的训练时间大幅减少,仅为原先的 20%左右。在测试集上的识别时间同样也有下降,但因其本身耗时较短,这样的改变影响不大。同时对比模型在测试集上的困惑度,发现加入类别层后的模型困惑度约 6%小幅度的升高,并未造成模型性能的大量下降。因此综合考虑,在模型训练时可以选择适当的类别层,减少模型训练的时间。

3.3.2　连续语音识别系统识别结果

在连续语音识别基线系统中,利用解码器从一次解码的结果中生成 Lattice,利用 Lattice-to-nbest 工具提取出 N-best 列表,并采用循环神经网络模型和 4-gram 模型对 N-best 列表进行重打分,得到最终的识别结果。其中循环神经网络语言模型的隐含层神经元个数为 200,类别层为 100,采用 BPTT 算法迭代 5 次,实验结果如表 3 所示。

从表 3 中可以看出,在二次解码中引入 4-gram 时,识别的词错误率的降低并不明显,对系统性能的提升有限。而通过 RNN 重打分算法引入 RNNLM,系统的词错误率从 16.31%降低到 15.51%,识别效果有明显的提高。这表明相比于 4-gram 模型,循环神经网络语言模型对语言现象的描述能力更强,具有更好的模型性能,对系统性能的提升也更大。引入缓存模型概率得分后,词错误率有进一步的下降,表明加入缓存模型可以避免因测试数据和训练数据不匹配造成模型性能的下降,从而提高系统的识别正确率。

表 3　RNN 模型 N-best 重打分对系统性能影响

Tab. 3　N-best rescoring performance using RNNLM　%

解码方法	词错误率	相对降低
基线系统(3-gram)	16.45	
4-gram 重打分	16.31	0.85
RNNLM 和 4-gram 重打分	15.51	5.71
RNNLM,4-gram 和缓存模型重打分	15.49	5.84

4　结束语

本文提出一种基于循环神经网络语言模型的 N-best 重打分算法,结合了高阶 n-gram 模型和循环神经网络语言模型的优势,并引入缓存模型对算法进行优化,使语言模型的概率得分计算更为精确,得到最优的识别结果。在微软语料库上的实验表明,本文方法能够降低语言模型的困惑度,并有效提高语音识别系统的正确率。同时,由于在语音解码过程中存在 N-best 重打分的重复计算,降低了解码的效率,系统识别速度仍有待改进。在下一步工作中将研究如何提高算法效率,以满足实时性要求较高的系统的需要。

参考文献：

[1] 王炳锡，屈丹，彭煊. 实用语音识别基础[M]. 北京：国防工业出版社，2005：287-291.
Wang Bingxi, Qu Dan, Peng Xuan. Practical fundamentals of speech recognition [M]. Beijing: National Defense Industry Press, 2005: 287-291.

[2] 荣传振，岳振军，贾永兴，等. 唇语识别关键技术研究进展[J]. 数据采集与处理，2012，27(S2)：277-283.
Rong Chuanzhen, Yue Zhenjun, Jia Yongxing, et al. Research advances in key technology of lip-reading [J]. Journal of Data Acquisition and Processing, 2012, 27(S2): 277-283.

[3] Rosenfeild R. Two decades of statistical language modeling: Where do we go from here? [J]. Proceedings of the IEEE, 2000, 88(8): 1270-1278.

[4] Sundermeyer M, Schluter R, Ney H. On the estimation of discount parameters for language model smoothing [C]// The 12th Annual Conference of the International Speech Communication Association. Florence, Italy: ISCA, 2011: 1433-1436.

[5] Deoras A, Mikolov T, Kombrink S, et al. Variational approximation of long-span language models for LVCSR [C]// IEEE International Conference on Acoustics, Speech and Signal Processing. Prague, Czech Republic: IEEE, 2011: 5532-5535.

[6] Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model [J]. Journal of Machine Learning Research, 2003, 3(2): 1137-1155.

[7] Mikolov T, Karafiat M, Burget L, et al. Recurrent neural network based language model [C]// The 11th International Speech Communication Association. Makuhari, Chiba, Japan: ISCA, 2010: 1045-1048.

[8] Sundermeyer M, Oparin I, Gauvain J L, et al. Comparison of feedforward and recurrent neural network language models [C]// IEEE International Conference on Acoustics, Speech and Signal Processing. Vancouver, Canada: IEEE, 2013: 8430-8434.

[9] Schwenk H. Continuous space language models [J]. Computer Speech and Language, 2007, 21(3): 492-518.

[10] Le H S, Oparin I, Allauzen A, et al. Structured output layer neural network language models for speech recognition [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2013, 21(1): 195-204.

[11] Mikolov T, Kombrink S, Burget L, et al. Extensions of recurrent neural network language model [C]// IEEE International Conference on Acoustics, Speech and Signal Processing. Prague, Czech Republic: IEEE, 2011: 5528-5531.

[12] 吕国云，赵荣椿，张燕宁，等. 基于三因素动态贝叶斯网络模型的大词汇量连续语音识别[J]. 数据采集与处理，2009，24(1)：1-6.
Lü Guoyun, Zhao Rongchun, Zhang Yanning, et al. Continuous speech recognition for large vocabulary based on triphone DBN model [J]. Journal of Data Acquisition and Processing, 2009, 24(1): 1-6.

[13] Kombrink S, Mikolov T, Karafiat M, et al. Recurrent neural network based language modeling in meeting recognition[C]// The 12th Annual Conference of the International Speech Communication Association. Florence, Italy: ISCA, 2011: 2877-2880.

[14] Povey D, Ghoshal A, Boulianne G, et al. The Kaldi speech recognition toolkit [C]// IEEE Automatic Speech Recognition and Understanding Workshop. Hawaii, USA: IEEE, 2011.

作者简介：

张剑(1988-)，男，硕士研究生，研究方向：语音识别和智能信息处理，E-mail：Crsmx_23@163.com。

屈丹(1974-)，通信作者，女，博士，副教授，研究方向：语音信号处理和模式识别，E-mail：qudanqudan@sina.com。

李真(1982-)，女，讲师，研究方向：语音识别和智能信息处理。