文章编号：1003-0077(2020)01-0063-08

基于稳健词素序列和 LSTM 的维吾尔语短文本分类

沙尔旦尔·帕尔哈提，米吉提·阿不里米提，艾斯卡尔·艾木都拉

（新疆大学 信息科学与工程学院，新疆 乌鲁木齐 830046）

摘 要：维吾尔语是一种派生类语言，其词是由词干和词缀连接而成的。其中，词干是有实际意义的词汇单元，词缀提供语法功能。该文提出了基于词干单元和长短期记忆(LSTM)网络的维吾尔语短文本分类技术。用基于词-词素平行训练语料的稳健词素切分和词干提取方法，从互联网下载的文本中提取其词干，以此构建词干序列文本语料库，并通过 Word2Vec 算法映射到实数向量空间。然后用 LSTM 网络作为特征选择和文本分类算法进行维吾尔语短文本分类实验，并得到 95.48%的分类准确率。从实验结果看，对于维吾尔语等派生类语言而言，特别是对于带噪声的文本，基于词干的分类方法有更多优异的性能。

关键词：维吾尔语；文本分类；LSTM；形态学

中图分类号：TP391 **文献标识码**：A

Uyghur Short Text Classification Based on Robust Morpheme Sequence and LSTM

SARDAR Parhat, MIJIT Ablimit, ASKAR Hamdulla

(School of Information Science and Engineering, Xinjiang University, Urumqi, Xinjiang 830046, China)

Abstract: Uyghur is a derivative language in which words are coined by stems concatenated with affixes, in which the stem is the word unit with practical meaning, and the affix provides grammatical function. This paper proposes Uyghur short text classification technique based on morpheme sequences and LSTM. A robust morpheme segmentation and stem extraction methods are trained on the word-morpheme parallel corpora to extract the stems from web texts. The stem sequence text corpus is thus obtained and then fed into the Word2Vec algorithm. With the achieved stem embedding, the LSTM is applied to implement Uyghur short text classification experiments. The experimental results show the proposed method achieves 95.48% classification accuracy, indicating that for derivative languages like Uyghur, especially for noisy texts, stem-based classification method has more excellent performance.

Keywords: Uyghur; text classification; LSTM; morphology

0 引言

近年来，随着互联网的快速发展，每天都会产生大量的文本、音频、图片和视频数据，其中文本信息的数据量最大。但文本信息混乱，难以人工区分和组织。因此，对文本数据进行自动分类已经成为一项紧迫的工作。

维吾尔语是个黏着性语言。维吾尔语的句子由自然分开的词组成。词由词干追加词缀来派生，因此词汇量巨大，其中词缀提供语义及语法功能，如表 1 所示。

表 1 维吾尔语词变体

词干	变体	词缀
（学校）mAktAp	（在学校）mAktAptA＝mAktAp＋tA	tA
	（学校的）mAktApniN＝mAktAp＋niN	niN

以上表中拉丁文字母对应的维吾尔文字母如表 2 所示。

收稿日期：2019-09-16 **定稿日期**：2019-09-22

基金项目：国家自然科学基金(61662078,61633013)；国家重点研发计划项目(2017YFC0820603)

表2 维吾尔语与拉丁字母对照表

序号	拉丁	维吾尔语	序号	拉丁	维吾尔语	序号	拉丁	维吾尔语
1	y	ي	12	p	پ	23	e	ې
2	a	ا	13	m	م	24	q	ق
3	l	ل	14	s	س	25	H	خ
4	G	غ	15	b	ب	26	U	ۇ
5	u	ۇ	16	d	د	27	h	ھ
6	z	ز	17	A	ە	28	g	گ
7	k	ك	18	v	ۋ	29	f	ف
8	x	ش	19	r	ر	30	w	ۋ
9	i	ى	20	n	ن	31	O	ۆ
10	t	ت	21	N	ڭ	32	J	ژ
11	o	و	22	c	ج	33	j	ج

维吾尔语中，词干是具有实际意义的词汇单元。词素切分及词干提取能够使我们获取有效的、有意义的特征，并降低特征的重复出现率和特征位数，如以下例子所示：

mAktAptA mAktApniN mAktAp baxqurux tUzUmi muhim.

中文意思：在学校，学校的[学校]管理纪律是重要的。

以上句子词素切分后变成：

mAktAp+tA mAktAp+niN mAktAp baxqur+ux tUzUm+i muhim.

以上句子中有6个词，其中前三个词的词干都是/mAktAp/(学校)，即一个词干能够表示三个词的主要意思，并获取三个词特征，特征位数会大幅减少。

维吾尔语自然语言处理(NLP)的主要问题是资源的缺乏和形态结构的变化，从因特网上收集的数据在编码和拼写等方面有带噪声和不确定性等特点[1]。方言以及在拼写和编码等方面的不确定性对提取和分类短和带有噪声的文本数据的可靠性带来了巨大挑战[2]。然而，提取和分类短、有噪声的文本数据是维吾尔语自然语言处理不可避免的重要步骤。

部分学者发表了维吾尔语词干提取有关的研究结果[3-4]。文献[3]根据简单的名词构形词缀规则进行构词成分有限状态分析，以此提取维吾尔语的词干。文献[4]根据维吾尔语的构词约束条件，用词性特征和上下文词干信息来提取维吾尔语词干。文献[4]没有考虑句子级别的上下文信息。以前的这些词干提取有关的工作大多是基于简单的后缀为基础的词干方法和一些简单的人工设置的规则。因此存在歧义，尤其是在短文本上。基于句子或长上下文的可靠词干提取方法可以正确预测噪声环境中的词干和词条，有利于维吾尔语等少数民族语言NLP的其他许多方面的研究。基于上述方法的多语种处理工具[2]可以为整个句子提供形态分析，并减少噪声文本中的歧义。有些学者对维吾尔文本分类做了一些研究[5-7]。文献[5]以KNN为分类器对维吾尔文本进行分类实验。在本研究中用词频-逆文档频率(TF-IDF)算法，来计算特征的权重值。文献[6]利用TextRank算法对维吾尔文本进行句子情感分类，文献[6]选用SVM作为分类算法。文献[7]用词频对传统的TF-IDF权值函数进行加权来改进，用贝叶斯分类器进行了维吾尔文本分类实验。

自然语言具有结构依赖的特点。以往的研究者在维吾尔文本分类中所使用的方法是在传统的分类框架下进行的，这些方法对文本中单词的频率和一些子词单元进行简单的统计，其中所使用的机器学习过程较浅，不考虑文本中词语之间的语义关系，因此无法保持文本上下文的清晰语义信息。

自动文本分类是一个引导性的学习过程，其中大量的非结构化文本信息(文本文档、网页等)根据给定的分类系统和文本信息的内容自动分类为指定的类别[8-9]。自动文本分类在许多信息检索工作和相关任务中得到广泛应用，包括情感分析[10]、垃圾邮件过滤[11]和观点挖掘[12]。

长短期记忆(LSTM)网络具有捕获顺序数据之间的依赖关系的能力，因为它善于捕获长距离信息，所以在语音识别中取得了显著的准确性[13-15]。LSTM网络可以有效地解决当用传统的分类模型进行自动文本分类时文本中词语上下文意义被忽视的问题。

文本表示和特征选择是文本挖掘和信息检索中的基本问题。它量化从文本中提取的特征词来表示文本信息。词袋模型(BOW)[16]和TF-IDF[17]常用于表示文本特征。本文中我们提出了基于词干单元的维吾尔短文本分类方法。采用基于形态规则的词素切分和词干提取方法从互联网收集的维吾尔文本语料库中稳健地提取其词干和用Word2Vec算法训练词干向量，并利用LSTM网络作为特征选择和文本分类算法，得到维吾尔文本分类模型后，进行分类实验。

1　维吾尔文本表示和分类方法

我们的分类算法主要包括两个部分。一是维吾尔文本集的预处理，包括对实验文本数据的获取、词素切分以及词干提取。二是分类过程，包括特征提取和分类。

1.1　词向量文本表示法

近期，深度神经网络和表示学习[18-19]提供了更好的文本表示方法和缓解数据稀疏问题。Mikolov 等[20]提出了 Word2Vec 文本表示方法，并利用深度学习和向量运算的思想，通过训练把文本内容的处理简化到 N 维向量空间，寻求文本数据更深层次的特征表示，并使用在向量空间中的相似性来表示文本的语义相似度。

词(词干)向量是一个实数向量[21]，通过计算两个给定的词干向量之间的距离，来得到它们的相似度。我们利用 Word2Vec 算法可以快速有效地训练词干向量。Word2Vec 算法包括两个重要的子模型：CBOW（连续词袋）模型[22] 和 Skip-gram 模型[23]。

CBOW 是一个在给定上下文词干 W_{t-c}，$W_{(t-c)-1},\cdots,W_{t-1},W_{t+1},W_{t+2},\cdots,W_{t+c}$ 的条件下预测特定词干 W_t 发生的概率 $P(W_t \mid W_{t-c},W_{(t-c-1)},\cdots,W_{t-1},W_{t+1},W_{t+2},\cdots,W_{t+c})$ 的模型。在这个模型中，一个词干由在这个词干前后的 c 个词干表示，c 是预选窗口的大小，输出是这个特征词干 W_t 的词干向量，如图 1 所示。我们将使用 CBOW 特征表示模型从噪声文本中得到词干向量。

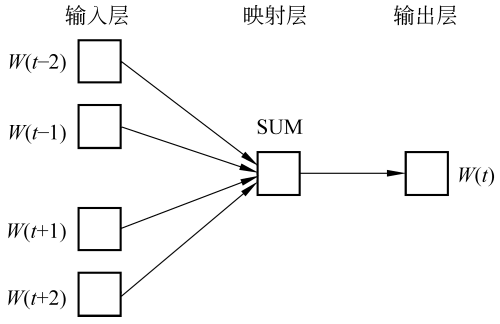

图 1　CBOW 模型

Skip-gram 模型的思想正好与 CBOW 模型相反，即，它在给定特定词干 W_t 的条件下，预测上下文词干 $W_{t-c},W_{(t-c-1)},\cdots,W_{t-1},W_{t+1},W_{t+2},\cdots,W_{t+c}$ 的发生概率 $P(W_{t-c},W_{(t-c)-1},\cdots,W_{t-1},W_{t+1},W_{t+2},\cdots,W_{t+c}\mid W_t)$，如图 2 所示。

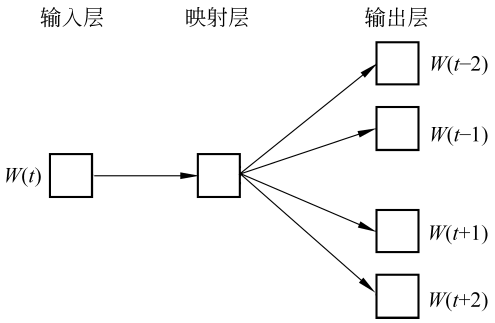

图 2　Skip-gram 模型

通过 Word2Vec 训练得到的词干向量可以通过其余弦距离来判断语义相似度。计算得到的余弦值越大，语义越相近；反之，语义相差越远，如表 3 所示。

表 3　词干向量语义相似度

词干 muzika(音乐)相关词干		词干 tor(网络)相关词干	
词干	余弦距离	词干	余弦距离
vusul(舞蹈)	0.813 8	bekAt(网站)	0.920 6
sAnvAt(艺术)	0.797 0	simsiz(无线)	0.878 3
naHxa(歌曲)	0.774 2	kOcmA(移动)	0.870 4
numur(表演)	0.749 5	soda(商务)	0.869 4
gitar(吉他)	0.741 3	vucur(信息)	0.842 7

从表 3 中可以看出，分别输入词干 muzika（音乐）和 tor(网络)，并通过计算词干向量之间的余弦距离，得到与这两个输入词干语义最相近的 5 个词干。

1.2　LSTM 网络框架

LSTM 网络是一种时间循环的神经网络，适用于处理和预测时间序列中间隔较长和延迟较大的重要事件[24]。LSTM 网络的一般架构如图 3 所示。每个节点包含三个门结构，分别为遗忘门、输入门和

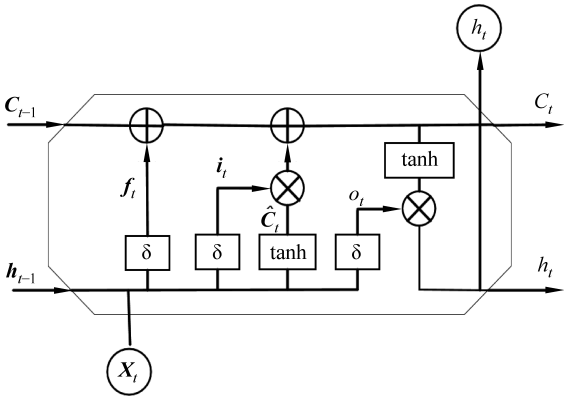

图 3　LSTM 网络结构

输出门。

在一个计算中，LSTM 节点首先计算遗忘门，如式(1)所示。

$$\boldsymbol{f}_t=\sigma(\boldsymbol{W}_f\cdot[h_{t-1},\boldsymbol{x}_t]+b_t) \tag{1}$$

其中，$\boldsymbol{W}_f$ 和 $\boldsymbol{x}_t$ 是遗忘门中 Sigmoid 层的参数矩阵和偏置矩阵，$\boldsymbol{f}_t$ 是遗忘向量，其中每一位都对应于 C_{t-1} 中的一个数据，用于控制信息的流入。

第二步是将输入数据 x_t 存储到节点中，并更新节点的状态。该步骤首先计算出第二个门结构，即输入门的 Sigmoid 层与一个 tanh 层，并得到两个选后值 i_t 和 $\overline{C}_t$，如式(2)和式(3)所示。

$$\boldsymbol{i}_t=\sigma(\boldsymbol{W}_i\cdot[\boldsymbol{h}_{t-1},\boldsymbol{x}_t]+b_i) \tag{2}$$

$$\overline{\boldsymbol{C}}_t=\tanh(\boldsymbol{W}_c\cdot[\boldsymbol{h}_{t-1},\boldsymbol{x}_t]+b_c) \tag{3}$$

然后，根据式(4)，得到节点的新状态 $\boldsymbol{C}_t$。

$$\boldsymbol{C}_t=f_t^*\boldsymbol{C}_{t-1}+\boldsymbol{i}_t*\overline{\boldsymbol{C}}_t \tag{4}$$

最后，节点通过第三个门结构，即输出门，获得输出向量，如式(5)所示。

$$o_t=\sigma(\boldsymbol{W}_0\cdot[\boldsymbol{h}_{t-1},\boldsymbol{x}_t]+b_0)$$
$$h_t=o_t*\tanh(\boldsymbol{C}_t) \tag{5}$$

通过增加门结构，LSTM 能够有效地避免传统的 RNN 模型训练困难这一问题，并有效地弥补在传统 RNN 网络上执行反向传播算法时梯度爆炸或者梯度消失等缺点。

1.3 稳健的维吾尔文本词素切分

由广泛的跨语言和跨文化交流所引起的书写形式上的不确定性，在给维吾尔文本带来噪声的同时，也导致新词、新概念和新表达的持续出现。这些新词大多是借用新进的外来词或词干，以及由于拼写习惯的不同和方言的变形而引起的噪声整合而成。书写形式上不确定性的另一个原因是书写系统的历史变化。这些不同的书写系统在现代社会留下了它们的遗产，虽然不太可能在官方媒体上出现，但却广泛存在于网上论坛和聊天工具中。

我们实验室开发的多语种处理工具[2]将单词序列切分成黏着性语言的词素序列，并且在功能和语言上都是可扩展的。

该工具根据词素和语音规则，从对齐的词-词素平行训练数据中自动学习黏着性语言词语的各种表面形式和声学变化。词素边界上的音素根据语音和谐规则改变其表面形式。当发音准确时，可以在文本中清楚地观察到语音和谐。该工具将导出每个候选词的所有可能的词素切分形式。将这些词素送入一个独立的统计模型，从前 N 个最好的词素中选择最佳词素。该工具为词干提取提供了可靠的依据，并极大地改进了短文本分类任务，词素切分流程如图 4 所示。

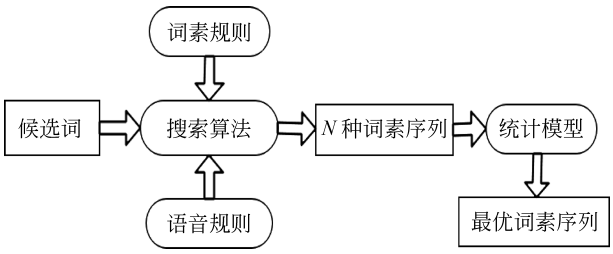

图 4 词素切分流程

该工具在包括 10 025 个句子的词-词素平行语料库上训练统计模型，其中选择了 9 025 个句子为训练语料，1 000 个句子为测试语料，做了词素切分和词干提取实验，其词干提取准确率最高时达到 97.66%。这是所有自动切分的词素与人工切分的词素完全匹配的百分比。文献[3]提出的方法词干提取准确率最高时达到 67%，比本文中的方法提取准确率低 30%。文献[4]中的词干提取方法是对我们采用的方法鲁棒性的提高。

表 4 给出了一些存在歧义的例子，这些例子只能通过句子等较长上下文中的形态学分析来消除其中的歧义，基于词内的词干提取方法无法可靠地提取词干。

表 4 有歧义的词干例子

变体	变体(拉丁文)	后缀
vAl(人民) / val(拿)	vAl/val→ vel+iN	iN
人的姓名/幸运的	qut→ qutluq, qut+luq	luq
火/草	vot→ vot+ tAk,vot+laq	tAk,laq

通常，实验语料库中原始文本的词长可能不一样。因此，我们应该使用填充来修改文本长度，以使所有文本具有相同的词长，从而产生 LSTM 所需的矩阵。我们统计了我们语料库中每个原始文本中的单词数量，如图 5 所示。(在图 5 中，横轴表示文本中的单词数量，纵轴表示对应某个单词量的文本数量)。

从图 5 可以看出，实验文本语料库中的文本词数趋向于约 40～120 词范围之内，而大部分文本约有 100 词左右。因此，我们选择了 100 个词作为用于 LSTM 的文本语料库的标准文本长度。我们用 0 填入词长不到 100 的文本。对于相应的词素序列文本，我们提取词干后为每个文本选择了前 100 个词干作

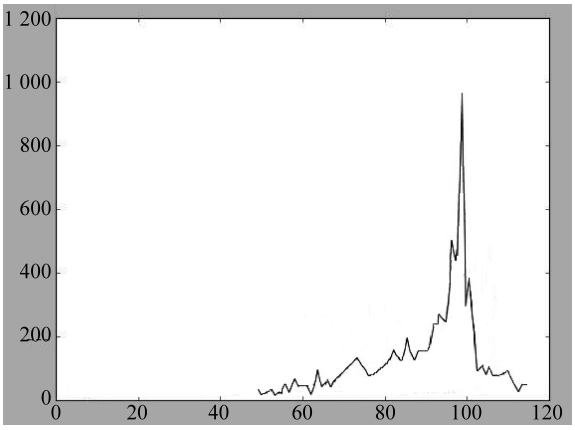

图 5　实验文本词长统计

为 LSTM 的输入，同样，我们用 0 填入词干数不到 100 的文本，以得到 LSTM 输入所需的文本矩阵。

2　实验结果与分析

目前，维吾尔文本的分类研究还处于起步阶段，尚无公开可用的标准的维吾尔文本语料库。因此，我们必须通过下载网上文本数据以构建维吾尔文本语料库，以此进行实验。

2.1　实验语料库

我们使用网络爬虫技术从官方的维吾尔文网，如 uyghur.people.com.cn，下载文本来构建我们的文本语料库。我们的语料库包括法律、金融、体育、文化、卫生、旅游、教育、科学与娱乐等 9 大类，每类包含 500 篇，共 4 500 篇。我们使用 75% 的文本（3 375 篇）作为训练文本语料，使用包括 450 篇的 10% 的文本作为验证语料，其余部分作为测试语料。

针对网络文本容易出现拼写错误的情况，我们开发了维吾尔文字拼写检查工具。该工具通过分析维吾尔语音节的结构形式和规则，可以发现大部分有拼写错误的词汇，从而使我们能够更正给定词汇中的拼写错误。拼写检查流程如图 6 所示。

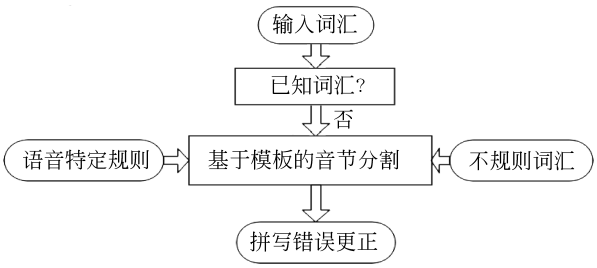

图 6　维吾尔文拼写检查流程

我们将所有文本从各种编码形式规范化成统一的罗马字母编码形式，并送入词素切分工具包中，转换成词素序列文本。基于词素和语音规则的词干提取方法能够很好地降低特征维数，其中词干词汇的数量显著地下降到词汇数量的 31% 以下，如表 5 所示。可以看出，随着类别和语料库数的增加，词干词汇的积累也只有词汇积累的 1/3。

表 5　词干提取引起的特征空间维数的减少

类别数	词汇数	词干数	词干-词汇比率/%
5	55 165	18 148	32.89
7	67 924	21 474	31.61
9	79 762	24 643	30.89

在稳健的词素切分和词干提取之后，用基于 Hierarchical Softmax 算法的 CBOW 模型分别训练所有语料库的词与词干向量，向量维度设置为 100，训练窗口设置为 5，学习速率设置为 0.025。

2.2　评价指标

精确率、召回率和 F_1 评分[25]用于评价文本分类的性能。其中，精确率和召回率反映了分类的两个方面，F_1 是二者的结合。计算公式如下：

精确率＝准确被分类到类别 C_k 的文本数量/实际被分类到类别 C_k 的所有文本数量

召回率＝准确被分类到类别 C_k 的文本数量/属于类别 C_k 的所有文本数量

F_1＝2×精确率×召回率/(精确率＋召回率)

对于我们所提出方法的评价，我们使用了宏观的 F_1 测度。宏观的 F_1 测度是一个全局的 F_1 指标。其中，首先分别计算每个类别的 F_1 得分，然后将这些 F_1 得分的算术平均值作为全局指标值。

2.3　实验环境

本文的实验环境如表 6 所示。

表 6　实验环境配置

实验环境	环境配置
操作系统	CentOS-7
处理器	GPU：1
内存	64GB
编程语言	Python 3.6
深度学习框架	Pytorch 1.0

2.4 实验结果与分析

我们将文本切分成词素序列并提取其词干，用Word2Vec对词干单元进行向量化，主要采用基于KNN[5]、NB[7]、SVM[6]、CNN和LSTM的分类方法进行了比较实验。前三种传统分类器上的分类实验中，用χ^2统计方法，根据词干项的CHI-2值大小选择了CHI-2值最大的前100到2 000之间的若干特征维数以实现特征降维，进行了分类实验，实验结果如表7所示。

表7 基于传统分类器的分类结果

CHI-2特征维数	特征	Word2Vec	
	分类器		
	KNN/%	NB/%	SVM/%
100	77.12	82.67	84.78
400	81.70	86.37	88.59
800	84.03	91.62	92.05
1 000	84.22	91.94	93.55
1 500	84.47	91.32	92.90
2 000	83.49	89.51	91.93

神经网络通过迭代计算获得权重，经多次迭代后得到理想的参数。基于CNN和LSTM的分类比较实验中，我们从所有的文本中分别提取100×100的词与词干两种向量，并分别送入CNN和LSTM。我们的实验中，为LSTM的层数选择了3个层，隐藏层的大小为64，用了交叉熵损失函数和Adam优化函数。我们将对基于词单元的分类结果与基于词干单元的分类结果进行了比较。我们做了150次迭代运算，如表8～表10所示。

从表7、8和9可以看出，特征维数在1 500和1 000时，基于KNN、NB和SVM的分类准确率最高分别达到84.47%、91.94%和93.55%。基于LSTM的实验中，在训练前段时间，随着迭代次数的增加，模型性能也随着增强，迭代次数达到40次左右时，模型基于词单元和词干单元的分类准确率都超过90%，并分别达到91.15%和93.57%。当迭代次数在60～70次左右时，迭代对模型性能的影响开始下降，模型训练到饱和状态，迭代次数达到90次和110次时，模型基于词干单元和词单元的分类准确率分别达到95.48%和93.76%等峰值后开始收敛。与基于三种传统分类方法的最高分类准确率相比，本文提出的方法最高分类准确率分别高出11.01%、3.54%和1.93%。基于词干单元的最高分类准确率比基于词单元的最高分类准确率高出1.72%。迭代次数增加时，训练时间也随着增加，但测试时间变化不大(给出了基于词干单元的训练和测试时间)。

表8 迭代次数对分类准确率的影响

实验性能	迭代次数									
	10	20	30	40	50	60	70	80	90	100
训练时间/s	1 021	2 690	5 622	7 752	12 513	17 891	27 610	32 495	37 451	42 128
测试时间/s	197	182	194	210	217	203	236	207	196	213
宏F_1(%)—词	83.44	85.93	88.27	91.15	92.47	92.94	93.18	93.11	93.22	93.48
宏F_1(%)—词干	86.73	89.96	92.62	93.57	94.89	95.19	95.23	95.33	95.48	95.11

表9 迭代次数对分类准确率的影响

实验性能	迭代次数				
	110	120	130	140	150
训练时间/s	47 150	52 640	57 093	61 724	67 629
测试时间/s	194	202	202	214	221
宏-F_1(%)—词	93.76	93.51	93.54	93.42	93.43
宏-F_1(%)—词干	95.23	95.25	95.36	95.28	95.19

表 10　基于 CNN 和 LSTM 的分类准确率比较

方法	词汇单元	宏 F_1 %	损失
CNN＋Word2Vec	词	93.28	0.24
	词干	95.07	0.16
LSTM＋Word2Vec	词	93.76	0.20
	词干	95.48	0.13

从表 10 可以看出，基于 LSTM 的分类准确率在词单元和词干单元上分别比基于 CNN 的分类准确率高出 0.48%和 0.41%。基于 LSTM 的分类损失值在词单元和词干单元上分别比基于 CNN 的分类损失值小于 4%和 3%。CNN 是在图像识别领域中有优势的网络结构，擅长于获取局部特征；LSTM 专注于前后关系的信息重建，能够有效地利用文本的上下文特征和顺序信息。通常，对于序列化的自然语言处理任务而言，LSTM 网络更有效率，但是由于我们的语料库规模较小，因此在与 CNN 的对比实验中，并没有显著地体现出 LSTM 模型在文本分类任务中的优越性。

3　结论

维吾尔语是一种形态丰富的黏着性语言，词是由词干加多个词缀所构成的，这一性质在理论上造成无限的词汇量。词缀提供语义和语法功能。因此，形态分析和词干提取是自然语言处理的有效途径。谷歌开发的词向量技术可以将语言单元映射成基于上下文的顺序向量空间。从上下文信息中提取和预测 OOV 是一种有效的方法。本文讨论了一种基于词—词素平行训练数据的稳健词素切分和词干提取方法，以及一种基于词干单元和神经网络结构的文本分类方法。基于 LSTM 网络的模型，维吾尔文本分类任务分别在词与词干单元上实现。实验结果表明，与词单元相比，词干单元有多种优良的性质，更合适派生类语言的处理。

参考文献

[1] Ablimit M, Kawahara T, Hamdulla A, et al. Stem-affix based Uyghur morphological analyzer[J]. International Journal of Future Generation Communication and Networking, 2016, 9 (2): 59-72.

[2] Ablimit M, Parhat S, Hamdulla A, et al. Multilingual language processing tool for Uyghur, Kazak and Kirghiz[C]//Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2017: 737-740.

[3] 早克热·卡德尔，艾山·吾买尔，艾斯卡尔·艾木都拉等. 维吾尔语名词构形词缀有限状态自动机的构造[J].中文信息学报，2009, 23(6): 116-122.

[4] 赛迪亚古丽·艾尼瓦尔，向露，艾斯卡尔·艾木都拉等. 融合多策略的维吾尔语词干提取方法[J]. 中文信息学报，2015, 29(5): 204-211.

[5] Tuerxun P, Fang D Y, Hamdulla A. The KNN based Uyghur text classification and its performance analysis [J]. International Journal of Hybrid Information Technology, 2015, 8 (3): 63-72.

[6] Iman S, Parhat R, Hamdulla A, et al. Performance analysis of different keyword extraction algorithms for emotion recognition from Uyghur text[C]//International Symposium on Chinese Spoken Language Processing. IEEE, 2014.

[7] 陈洋，哈力旦·阿布都热依木，伊力亚尔·达吾提，等. 基于加权改进贝叶斯算法的维吾尔文文本分类[J]. 计算机工程与设计，2014, 35(6): 1999-2003.

[8] Mccallum A, Nigam K. Text classification by bootstrapping with keywords, EM and shrinkage [C]//Proceedings of the ACL99-Workshop for Unsupervised Learning in Natural Language Processing, 1999: 51-58.

[9] Zhang Y, Zincirheywood N, Millos E. Narrative text classification for automatic key phrase extraction in web document corpora[C]//Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management, 2005: 52-58.

[10] 王汝娇，姬东鸿. 基于卷积神经网络与多特征融合的 Twitter 情感分类方法[J]. 计算机工程，2018, 44 (2): 210-219.

[11] Zhou B, Yao Y, Luo J. Cost-sensitive three-way email spam filtering[J]. Journal of Intelligent Information Systems, 2014, 42 (1): 19-45.

[12] Aggarwal C C, Zhai C. A survey of text classification algorithms[J]. In Mining text data, 2012: 163-222.

[13] Xi X F, Dong Z G. A survey on deep learning for natural language processing[J]. Acta Automatica Sinica, 2016, 42 (10): 1445-1465.

[14] 赵淑芳，董小雨. 基于改进的 LSTM 深度神经网络语音识别研究[J]. 郑州大学学报(工学版)，2018, 39 (05): 67-71.

[15] 张宇，张鹏远，颜永红，等. 基于注意力 LSTM 和多任务学习的远场语音识别[C]//第十四届全国人机语音通讯学术会议(NCMMSC'2017)论文集，2017.

[16] Wallac H, Hanna M. Topic modeling: Beyond bag-of-words[C]//Proceedings of the International Conference on Machine Learning. ACM, 2006: 977-984.

[17] Hu J, Yao Y. Research on the application of an improved TFIDF algorithm in text classification[J]. Journal of Convergence Information Technology, 2013, 8(7): 639-646.

[18] Bengio Y, Schwent H, Senécal J S, et al. Neural probabilistic language models[M]//Innovations in Machine Learning. Springer Berlin Heidelberg, 2006: 211-219.

[19] Mnih A, Hinton G. Three new graphical models for statistical language modelling[C]//Proceedings of 24th International Conference on Machine Learning. ACM, 2007: 641-648.

[20] Mikolov T, Sutskever T, Chen K, et al. Distributed representation of words and phrases and their compositionality[C]//Proceedings of Advances in Neural Information Processing Systems, 2013, 26: 3111-3119.

[21] Lai S W, Xu L H, Kang L, et al. Recurrent convolutional neural networks for text classification[C]//Proceedings of the 29th AAAI Conference on Artificial Intelligence, 2015: 2267-2273.

[22] Goldberg Y, Levy O. Word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method[J]. Eprint Arxiv.1402.3722, 2014.

[23] Chen Y Q, Nixon M S, Damper R I. Implementing the k-nearest neighbour rule via a neural network[C]//Proceedings of the IEEE International Conference on Neural Networks. IEEE, 1995: 136-140.

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

[25] Sebastian I, Fabrizi O. Machine learning in automated text categorization[J]. ACM Computing Surveys, 2002, 34 (1): 1-47.

沙尔旦尔·帕尔哈提(1984—),博士研究生,主要研究领域为文本及图像信息检索。
E-mail: sardar312@126.com

艾斯卡尔·艾木都拉(1972—),通信作者,博士,教授,主要研究领域为图像处理与模式识别、智能信息检索。
E-mail: askar@xju.edu.cn

米吉提·阿不里米提(1974—),博士,教授,主要研究领域为语音和语言信息处理。
E-mail: mijit@xju.edu.cn