计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2020.20190522
57(7): 1481-1489, 2020

基于扩展的 S-LSTM 的文本蕴含识别

胡超文　邬昌兴　杨亚连
(华东交通大学软件学院　南昌　330013)
(hcwjuly@126.com)

Extended S-LSTM Based Textual Entailment Recognition

Hu Chaowen, Wu Changxing, and Yang Yalian
(*School of Software, East China Jiaotong University, Nanchang* 330013)

Abstract　Text entailment recognition aims at automatically determining whether there is an entailment relationship between the given premise and hypothesis (usually two sentences). It is a basic and challenging task in natural language processing. Current dominant models, which are based on deep learning, usually encode the semantic representations of two sentences separately, instead of considering them as a whole. Besides, most of them do not leverage both the sentence-level global and ngram-level local information when capturing the semantic relationship. The recently proposed S-LSTM can learn semantic representations of a sentence and its ngrams simultaneously, achieving promising performance on tasks such as text classification. Considering the above, a model based on an extended S-LSTM is proposed for textual entailment recognition. On the one hand, S-LSTM is extended to learn semantic representations of the premise and hypothesis simultaneously, which regards them as a whole. On the other hand, to obtain better semantic representation, both the sentence-level and ngram-level information are used to capture the semantic relationships. Experimental results, on the English SNLI dataset and Chinese CNLI dataset, show that the performance of the proposed model is better than baselines.

Key words　textual entailment recognition; entailment relationship; semantic relationship; extended S-LSTM; deep learning

摘　要　文本蕴含识别旨在自动判断给定的前提和假设(通常为 2 个句子)之间是否存在蕴含关系,是自然语言处理领域一项基础但富有挑战的任务.当前,主流的基于深度学习的模型通常分别建模前提和假设的语义表示,而没有把它们看作一个整体;另外,在捕获它们之间的语义关系时,大都没有同时利用句子级别的全局信息和短语级别的局部信息.最近提出的 S-LSTM 能够同时学习句子和短语的语义表示,在文本分类等任务上取得了较好的效果.基于上述情况,提出了一种基于扩展的 S-LSTM 的文本蕴含识别模型.一方面,把前提和假设看作一个整体,扩展 S-LSTM 以同时学习它们的语义表示;另一方面,在建模语义关系时,既利用句子级别的信息又利用短语级别的信息,以此获得更好的语义表示.在英文 SNLI 数据集和中文 CNLI 数据集上的实验结果表明:提出的模型取得了比基准模型更好的识别性能.

收稿日期:2019-08-12;**修回日期**:2020-02-13
基金项目:国家自然科学基金项目(61866012);江西省自然科学基金项目(20181BAB202012);江西省教育厅科学技术研究项目(GJJ180329)
This work was supported by the National Natural Science Foundation of China (61866012), the Natural Science Foundation of Jiangxi Province of China (20181BAB202012), and the Science and Technology Research Project of Jiangxi Provincial Education Department (GJJ180329).
通信作者:邬昌兴(wcxnlp@163.com)

关键词 文本蕴含识别;蕴含关系;语义关系;扩展的S-LSTM;深度学习

中图法分类号 TP391

文本蕴含关系(textual entailment)作为一种基本的文本间语义关系,广泛存在于自然语言文本中.文本蕴含识别(textual entailment recognition, TER)是自然语言处理领域的一项基础性工作,其识别性能的提高将促进许多下游自然语言处理应用的发展.例如在问答系统中用来生成候选答案、文档摘要中用来辅助精简文本、机器翻译中用来评估翻译系统的性能[1].文本蕴含识别通常被看作一个分类任务,即给定前提与假设(通常为2个句子),自动判断它们之间是否存在蕴含关系或矛盾关系.如表1所示,在常用的中文CNLI①(Chinese natural language inference)和英文SNLI[2](stanford natural language inference)数据集中,定义了前提与假设之间的3种关系:蕴含、中立和矛盾.

Table 1 Examples in CNLI

表1 CNLI数据集中的例子

Premise	Hypothesis	Label
女孩子愉快地在沙滩上寻找贝壳.	有人在户外活动.	Entailment
一条黑色的狗穿过田野.	狗正在追逐一只猫.	Neutral
在白色的雪中挖掘的黑狗.	那只白狗正在土里挖一个洞.	Contradiction

文本蕴含识别同时也是一项非常具有挑战性的任务.有的情况下可以基于2个句子中具有矛盾关系的词或短语对进行推断,如表1矛盾示例中的"黑狗"与"白狗";而有些情况下则需要理解2个句子的语义才能进行准确的判断,如表1中的蕴含示例和中立示例.因此,文本蕴含识别既需要句子级别的全局信息,也需要短语级别的局部信息.

早期的文本蕴含识别方法主要包括:基于规则的方法[3]、基于相似度的方法[4]、基于对齐特征或其他人工定义特征的机器学习方法[5-6].这些早期的方法由于不能很好地对句子的语义进行建模,识别的性能并不理想.近年来,基于深度学习的方法在语义建模方面取得了很好的效果.例如基于双向长短时记忆网络(bidirectional long short-term memory network, Bi-LSTM)学习句子语义表示的模型在句法分析、机器翻译和实体关系抽取等诸多自然语言任务上取得了当前最好的效果[7-9].就文本蕴含识别而言,基于深度学习方法的性能已经全面超越早期的方法,成为当前主流的文本蕴含识别方法.

现有基于深度学习的文本蕴含识别方法可大致归为2类:基于句子编码的方法[2,10-17]和基于短语交互的方法[18-23].前者通常首先利用深度神经网络(例如Bi-LSTM、卷积神经网络CNN等)分别学习前提和假设的语义向量表示,然后推导它们之间的语义关系.后者通常首先分别学习前提和假设中词和短语(ngram)在上下文中的语义向量表示,并建模这些局部信息之间的语义关系,继而推断前提和假设之间的全局语义关系.当前基于深度学习的方法虽然取得了较好的识别性能,但仍然具有2点不足之处:

1) 分别学习前提和假设(或其中的词和短语)的语义向量表示,而没有把它们当作一个整体.直觉上,人类在进行蕴含关系推断时,会来回阅读2个句子.通过它们之间的信息交换,以达到充分理解句子语义的目的.

2) 没有同时利用句子级别的全局信息和短语级别的局部信息.例如基于句子编码的方法主要利用了句子级别的信息,而基于短语交互的方法则主要利用了词和短语级别的信息.

最近提出的S-LSTM(sentence-state LSTM)是一种能有效地建模文本序列的神经网络模型[24].S-LSTM通过迭代的方式在句子和短语之间交换信息,以同时学习句子级别的全局语义表示和短语级别的局部语义表示,在多个文本分类和序列标注任务上取得了优于Bi-LSTM的效果.本文提出一种基于扩展的S-LSTM的文本蕴含识别模型.具体地,我们把前提和假设看作一个整体,扩展S-LSTM以同时学习它们的语义表示.也就是说,在学习前提(假设)及其短语的语义向量表示时,考虑假设(前提)的语义信息.另一方面,在建模前提和假设之间的语义关系时,既考虑句子级别的全局信息,也考虑词和短语级别的局部信息.

在常用的英文SNLI数据集和中文CNLI数据集上的实验结果表明,本文提出的方法与基于句子编码或基于短语交互的基准方法相比,识别性能取得了一定的提高.

① https://github.com/blcunlp/CNLI

1 相关工作

借助于深度学习的发展和大规模数据集 SNLI 的发布,基于深度学习的文本蕴含识别方法成为当前的研究热点之一,其性能已经全面超越早期的方法.最近的研究工作可大致分为 2 类:基于句子编码的方法和基于短语交互的方法.

1) 基于句子编码的方法首先通过 Bi-LSTM 等神经网络分别学习前提和假设的语义向量表示,然后使用拼接、内积和作差等简单操作建模它们之间的语义关系[2,10-17].例如 Bowman 等人[2]基于 LSTM 从左至右学习前提和假设的语义表示;Liu 等人[11]使用 Bi-LSTM 网络从左至右和从右至左 2 个方向加强语义的学习和表示;Chen 等人[12]则使用多层 Bi-LSTM 网络学习前提和假设的层次化的语义表示,并基于池化操作(pooling)提取显著的特征.为了利用句子的结构信息;Mou 等人[13]使用一种基于依存树结构的卷积神经网络,并取得了较好的识别性能;谭咏梅等人[14]联合使用 CNN 和 Bi-LSTM 网络,以充分发挥 CNN 利于局部信息建模和 Bi-LSTM 利于全局信息建模的优点.最近 Shen 等人[15]使用自注意力(self-attention)机制代替以前常用的 Bi-LSTM 和 CNN,以发挥其能够捕获任意距离词之间的依赖的优势,并通过 Mask 矩阵加入方向信息.这类方法强调如何学习前提和假设的语义向量表示,而通常使用较简单的操作建模语义关系.

2) 基于短语交互的方法首先分别学习前提和假设中词和短语在上下文中的语义向量表示,然后引入注意力机制(attention mechanism)捕获这些局部信息之间的语义关系,继而推断前提和假设之间的全局语义关系[18-23].例如 Parikh 等人[18]将蕴含关系识别问题分解成词之间的对齐问题,利用双向注意力机制直接基于词的语义向量表示建模前提和假设之间的关系;Chen 等人[20]首先基于 Bi-LSTM 编码词在上下文中的词义表示,然后基于双向注意力机制(bi-attention mechanism, BiAttn)计算局部语义信息,最后把这些局部信息输入到另一个 Bi-LSTM 计算全局蕴含关系;Tan 等人[21]融合了 4 种不同的双向注意力机制的计算方法,用于加强局部语义关系的计算.这类方法强调如何显式地建模词之间或短语之间的交互,以有效地捕获局部语义关系.

本文提出的方法可以看作是基于句子编码方法和短语交互方法的结合与改进,主要体现在 2 方面:1)通过扩展 S-LSTM 网络,同时学习前提和假设的语义向量表示.这一点可以看作是对基于句子编码方法的改进,即学习了更好的句子级别的信息.2)在建模语义关系时,同时考虑句子级别和短语级别的信息,这可以看作是对基于短语交互方法的改进.另外,本文也是首次把 S-LSTM 网络应用到文本蕴含识别中,取得了比常用的 Bi-LSTM 更好的效果.

2 基于扩展的 S-LSTM 的文本蕴含识别模型

基于扩展的 S-LSTM 的文本蕴含识别模型以前提和假设为输入,输出它们之间是“蕴含”、“中立”还是“矛盾”关系.如图 1 所示,该模型包括 5 层:词向量层、编码层、交互层、聚合层和 MLP(multilayer perceptron)层.

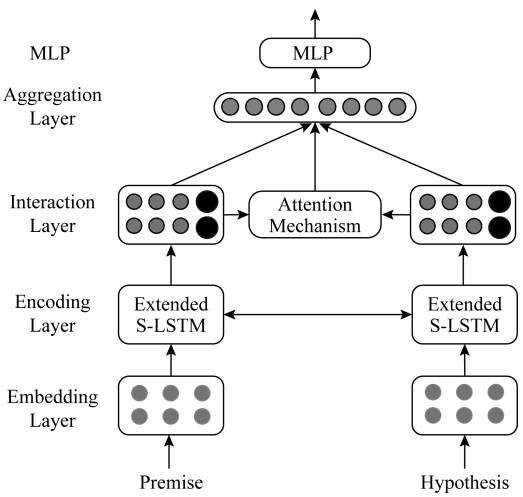

Fig. 1 Extended S-LSTM based textual entailment recognition model

图 1 基于扩展的 S-LSTM 的文本蕴含识别模型

词向量层把词编码成向量表示;编码层基于扩展的 S-LSTM 学习短语级别的信息和句子级别的信息;交互层基于双向注意力机制学习局部的语义关系表示;聚合层用于学习全局的语义关系表示;MLP 层用于输出最终的分类结果.图 1 编码层中双向箭头和交互层中黑色大圆圈部分为本文的 2 处改进,即编码层前提和假设之间的信息交换和交互层中句子级别信息的利用.下面各节分别对每一层进行详细的介绍.

2.1 词向量层

基于神经网络的自然语言处理模型通常先把词表示为向量,用作模型的输入.给定基于大规模语料预训练好的词向量(word embeddings)$\boldsymbol{E}\in\mathbb{R}^{|V|\times d}$,

其中,$|V|$表示词表的大小,d 为词向量的维度,$\boldsymbol{E}$ 中的每一行即为词表中对应词的语义向量表示.词向量层把前提和假设(可看作词序列)表示为向量序列,分别记作 $\boldsymbol{A}=(a_1,a_2,\cdots,a_i,\cdots,a_m)$和 $\boldsymbol{B}=(b_1,b_2,\cdots,b_j,\cdots,b_n)$,其中,$a_i,b_j\in\mathbb{R}^d$ 分别为对应词的向量表示,m 和 n 分别为前提和假设的长度.大量的研究工作证实预训练的词向量中包含语义信息,例如"高兴"和"快乐"这 2 个词在语义向量空间比较接近,非常适合用作自然语言模型的输入.

2.2 编码层

以前提和假设的词向量表示作为输入,编码层用于计算它们的语义向量表示.具体地,利用本文提出的扩展的 S-LSTM(详见第 4 节)把前提和假设分别表示为

$$\begin{aligned}\boldsymbol{H}_a&=(\boldsymbol{h}_{a_1},\cdots,\boldsymbol{h}_{a_i},\cdots,\boldsymbol{h}_{a_m},\boldsymbol{g}_a)^{\mathrm{T}},\\ \boldsymbol{H}_b&=(\boldsymbol{h}_{b_1},\cdots,\boldsymbol{h}_{b_i},\cdots,\boldsymbol{h}_{b_n},\boldsymbol{g}_b)^{\mathrm{T}},\end{aligned}\tag{1}$$

其中,$\boldsymbol{h}_{a_i}$ 可以看作是前提中以第 i 个词为中心的短语(ngram)的语义向量表示,是局部的语义信息;$\boldsymbol{g}_a$ 是前提的语义向量表示,是全局的语义信息;类似地,$\boldsymbol{h}_{b_j}$ 是假设中短语的语义表示,$\boldsymbol{g}_b$ 是假设的全局语义表示;$\boldsymbol{H}_a\in\mathbb{R}^{(m+1)\times d}$ 和 $\boldsymbol{H}_b\in\mathbb{R}^{(n+1)\times d}$ 为编码层的输出.

与常用的 Bi-LSTM 相比,扩展的 S-LSTM 具有 2 个方面的优点:

1) 如图 1 编码层中双向箭头连线所示,在学习前提中短语的向量表示 $\boldsymbol{h}_{a_i}$ 时,会考虑假设的语义 $\boldsymbol{g}_b$,反之亦然,具体可参见第 4 节式(6).

2) 如图 1 交互层中黑色大圆圈所示,在利用短语表示的同时,还利用了前提和假设在句子级别的语义向量表示 $\boldsymbol{g}_a$ 和 $\boldsymbol{g}_b$.理论上,Bi-LSTM 中最后一个词对应的表示也可以看作是整个句子的语义表示.然而,当句子较长时,Bi-LSTM 难以捕获词之间长距离的依赖信息.因此,多数情况下其最后一个词对应的向量表示实质上仅是短语级别的信息.

2.3 交互层

交互层采用双向注意力机制(BiAttn)[18]捕获前提和假设之间的局部语义关系.具体地,首先计算 $\boldsymbol{Q},\overline{\boldsymbol{H}}_a,\overline{\boldsymbol{H}}_b$:

$$\begin{aligned}\boldsymbol{Q}&=F(\boldsymbol{H}_a)F(\boldsymbol{H}_b)^{\mathrm{T}},\\ \overline{\boldsymbol{H}}_a&=softmax(\boldsymbol{Q}^{\mathrm{T}})\boldsymbol{H}_a,\\ \overline{\boldsymbol{H}}_b&=softmax(\boldsymbol{Q})\boldsymbol{H}_b,\end{aligned}\tag{2}$$

其中,F 是多层前馈神经网络;$\boldsymbol{Q}\in\mathbb{R}^{(m+1)\times(n+1)}$ 是相关性矩阵,$\boldsymbol{Q}$ 中元素 q_{ij} 是前提中 $\boldsymbol{h}_{a_i}$ 与假设中 $\boldsymbol{h}_{b_j}$ 的关联程度;$softmax(\boldsymbol{Q}^{\mathrm{T}})$ 和 $softmax(\boldsymbol{Q})$ 分别表示对 $\boldsymbol{Q}$ 按列和行进行归一化;$\overline{\boldsymbol{H}}_a\in\mathbb{R}^{(n+1)\times d}$ 中第 j 行可认为是前提中与 $\boldsymbol{h}_{b_j}$ 相关联内容的语义向量表示,类似地有 $\overline{\boldsymbol{H}}_b\in\mathbb{R}^{(m+1)\times d}$.然后,计算局部语义关系的向量表示 $\boldsymbol{V}_a\in\mathbb{R}^{(m+1)\times d_1}$ 和 $\boldsymbol{V}_b\in\mathbb{R}^{(n+1)\times d_1}$:

$$\begin{aligned}\boldsymbol{V}_a&=G([\boldsymbol{H}_a;\overline{\boldsymbol{H}}_b]),\\ \boldsymbol{V}_b&=G([\boldsymbol{H}_b;\overline{\boldsymbol{H}}_a]),\end{aligned}\tag{3}$$

其中,G 是多层前馈神经网络,d_1 是 G 中最后一层的维度,[;]表示向量的拼接操作.

在局部语义关系的计算过程中,既有短语表示 $\boldsymbol{h}_{a_i}$ 与 $\boldsymbol{h}_{b_j}$ 之间的交互,又有短语表示 $\boldsymbol{h}_{a_i}$($\boldsymbol{h}_{b_j}$)与句子表示 $\boldsymbol{g}_b$($\boldsymbol{g}_a$)之间的交互,还有句子表示 $\boldsymbol{g}_a$ 与 $\boldsymbol{g}_b$ 之间的交互.可以说,通过句子级信息的引入,交互层更好地建模了前提和假设之间的局部语义关系.

2.4 聚合层

聚合层在局部语义关系的基础上计算全局语义关系表示,采用与文献[20]中类似的方法:

$$\begin{aligned}\boldsymbol{v}_a&=[\max(\boldsymbol{V}_a);avg(\boldsymbol{V}_a)],\\ \boldsymbol{v}_b&=[\max(\boldsymbol{V}_b);avg(\boldsymbol{V}_b)],\\ \boldsymbol{o}&=[\boldsymbol{v}_a;\boldsymbol{v}_b;\boldsymbol{v}_a\cdot\boldsymbol{v}_b;|\boldsymbol{v}_a-\boldsymbol{v}_b|],\end{aligned}\tag{4}$$

其中,$\boldsymbol{v}_a,\boldsymbol{v}_b\in\mathbb{R}^{2d_1}$;max 和 avg 分别表示最大池化和平均池化操作;$\boldsymbol{v}_a\cdot\boldsymbol{v}_b$ 表点乘,$|\boldsymbol{v}_a-\boldsymbol{v}_b|$表示作差后取绝对值;$\boldsymbol{o}\in\mathbb{R}^{8d_1}$ 即为全局语义关系表示.

2.5 MLP 层

MLP 层由多个非线性隐层和一个 $softmax$ 层组成,用于计算最终的分类结果:

$$\bar{y}=C_{\mathrm{MLP}}(\boldsymbol{o}),\tag{5}$$

其中,$\bar{y}\in\mathbb{R}^3$ 中每一维即为相应类别的概率.

3 扩展的 S-LSTM

扩展的 S-LSTM 的网络结构如图 2 所示,其基本思想是把常用于单个句子建模的 S-LSTM[24]扩展用于处理句对的情况.核心是同时编码 2 个句子,并考虑它们之间信息的交换.具体地,图 2 中左半部分是对前提的建模,右半部分是对假设的建模,假设到前提的连线(为了保持简洁,省略了前提到假设的连线)体现了它们之间信息的交换.

扩展的 S-LSTM 基于迭代的方式计算句子和短语的语义表示.在任一时刻 t,扩展的 S-LSTM 把前提表示为 2 类信息:短语级别的信息 $\boldsymbol{h}_{a_i}^t$ 和句子级别的信息 $\boldsymbol{g}_a^t$.给定时刻 $t-1$ 前提的语义表示为 $\boldsymbol{H}_a^{t-1}=(\boldsymbol{h}_{a_1}^{t-1},\boldsymbol{h}_{a_2}^{t-1},\cdots,\boldsymbol{h}_{a_i}^{t-1},\cdots,\boldsymbol{h}_{a_m}^{t-1},\boldsymbol{g}_a^{t-1})$,首先按如下方式计算时刻 t 前提中的短语语义表示 $\boldsymbol{h}_{a_i}^t$(如图 2 中实线箭头所示):

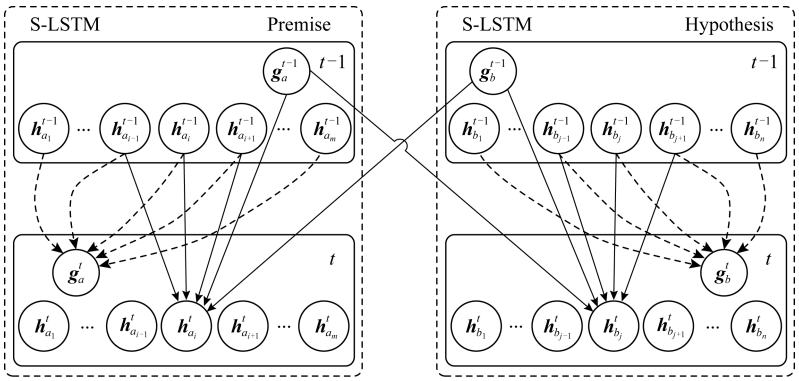

Fig. 2 Extended S-LSTM

图 2 扩展的 S-LSTM

$$
\begin{aligned}
\boldsymbol{\xi}_i^t &= (\boldsymbol{h}_{a_{i-1}}^{t-1}, \boldsymbol{h}_{a_i}^{t-1}, \boldsymbol{h}_{a_{i+1}}^{t-1}),\\
\hat{\boldsymbol{i}}_i^t &= \sigma(\boldsymbol{W}_i\boldsymbol{\xi}_i^t + \boldsymbol{U}_i\boldsymbol{a}_i + \boldsymbol{V}_a^i\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^i\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_i),\\
\hat{\boldsymbol{l}}_i^t &= \sigma(\boldsymbol{W}_l\boldsymbol{\xi}_i^t + \boldsymbol{U}_l\boldsymbol{a}_i + \boldsymbol{V}_a^l\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^l\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_l),\\
\hat{\boldsymbol{r}}_i^t &= \sigma(\boldsymbol{W}_r\boldsymbol{\xi}_i^t + \boldsymbol{U}_r\boldsymbol{a}_i + \boldsymbol{V}_a^r\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^r\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_r),\\
\hat{\boldsymbol{f}}_i^t &= \sigma(\boldsymbol{W}_f\boldsymbol{\xi}_i^t + \boldsymbol{U}_f\boldsymbol{a}_i + \boldsymbol{V}_a^f\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^f\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_f),\\
\hat{\boldsymbol{s}}_i^t &= \sigma(\boldsymbol{W}_s\boldsymbol{\xi}_i^t + \boldsymbol{U}_s\boldsymbol{a}_i + \boldsymbol{V}_a^s\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^s\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_s),\\
\boldsymbol{o}_i^t &= \sigma(\boldsymbol{W}_o\boldsymbol{\xi}_i^t + \boldsymbol{U}_o\boldsymbol{a}_i + \boldsymbol{V}_a^o\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^o\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_o),\\
\boldsymbol{u}_i^t &= \tanh(\boldsymbol{W}_u\boldsymbol{\xi}_i^t + \boldsymbol{U}_u\boldsymbol{a}_i + \boldsymbol{V}_a^u\boldsymbol{g}_a^{t-1} + \boldsymbol{V}_b^u\boldsymbol{g}_b^{t-1} + \boldsymbol{b}_u),\\
\boldsymbol{i}_i^t, \boldsymbol{l}_i^t, \boldsymbol{r}_i^t, \boldsymbol{f}_i^t, \boldsymbol{s}_i^t &= softmax(\hat{\boldsymbol{i}}_i^t, \hat{\boldsymbol{l}}_i^t, \hat{\boldsymbol{r}}_i^t, \hat{\boldsymbol{f}}_i^t, \hat{\boldsymbol{s}}_i^t),\\
\boldsymbol{c}_{a_i}^t &= \boldsymbol{l}_i^t \odot c_{a_{i-1}}^{t-1} + \boldsymbol{f}_i^t \odot c_{a_i}^{t-1} + \boldsymbol{r}_i^t \odot c_{a_{i+1}}^{t-1} +\\
&\quad \boldsymbol{s}_i^t \odot c_{\boldsymbol{g}_a}^{t-1} + \boldsymbol{i}_i^t \odot \boldsymbol{u}_i^t,\\
\boldsymbol{h}_{a_i}^t &= \boldsymbol{o}_t^i \odot \tanh(\boldsymbol{c}_{a_i}^t),
\end{aligned}
\tag{6}
$$

其中,$\boldsymbol{\xi}_i^t$ 是窗口大小的局部信息,$\boldsymbol{i}_i^t, \boldsymbol{l}_i^t, \boldsymbol{r}_i^t, \boldsymbol{f}_i^t, \boldsymbol{s}_i^t, \boldsymbol{o}_i^t$ 都是用于控制信息流的门(gate).具体地,$\boldsymbol{l}_i^t, \boldsymbol{r}_i^t, \boldsymbol{f}_i^t, \boldsymbol{s}_i^t$ 是遗忘门,用于遗忘部分时刻 $t-1$ 的信息;$\boldsymbol{i}_i^t$ 是输入门,用于提取当前位置的输入信息 $\boldsymbol{u}_i^t$;$\boldsymbol{o}_i^t$ 为输出门,用于控制当前位置的输出信息.$\boldsymbol{c}_{a_i}^t$ 是对应于 $\boldsymbol{h}_i^t$ 的记忆单元,类似的记忆单元包括 $\boldsymbol{c}_{a_{i-1}}^{t-1}, \boldsymbol{c}_{a_i}^{t-1}, \boldsymbol{c}_{a_{i+1}}^{t-1}, \boldsymbol{c}_{\boldsymbol{g}_a}^{t-1}$.模型的参数设置为 $\boldsymbol{W}_x, \boldsymbol{U}_x, \boldsymbol{V}_a^x, \boldsymbol{V}_b^x, \boldsymbol{b}_x$ ($x \in \{\boldsymbol{i}, \boldsymbol{o}, \boldsymbol{l}, \boldsymbol{r}, \boldsymbol{f}, \boldsymbol{s}, \boldsymbol{u}\}$),$\sigma$ 表示 $sigmoid$ 函数.然后,以短语级别的信息 $\boldsymbol{h}_{a_i}^t$ 为输入,计算时刻 t 的前提语义表示 $\boldsymbol{g}_a^t$(如图 2 左半部分中虚线箭头所示).$\boldsymbol{g}_a^t$ 的计算过程与 S-LSTM 中的相同,这里不再赘述,详情可参见文献[24]中的式(3).类似地,可计算得到假设在时刻 t 的语义表示 $\boldsymbol{h}_{b_j}^t$ 和 $\boldsymbol{g}_b^t$.

扩展的 S-LSTM 与原始的 S-LSTM 的主要不同之处在于:对前提进行建模时,考虑了假设的语义表示 $\boldsymbol{g}_b^{t-1}$,如式(6)中 $\boldsymbol{i}_i^t, \boldsymbol{l}_i^t, \boldsymbol{r}_i^t, \boldsymbol{f}_i^t, \boldsymbol{s}_i^t, \boldsymbol{u}_i^t$ 的计算.类似地,在对假设进行建模时,也考虑了前提的语义表示.与前人工作中对前提和假设分别进行建模的方式相比,扩展的 S-LSTM 把前提和假设看作一个整体,通过它们之间的信息交换,学习更好的语义表示.

4 实 验

4.1 数据与设置

我们在常用的英文 SNLI 数据集和中文 CNLI 数据集上验证所提方法的有效性.SNLI 原始数据集包含 570 152 个句子对,每个句子对使用以下关系标记:蕴含、矛盾、中立和"—",其中"—"表示人类标注者缺乏共识而最终没有给定标签.为了进行公平的比较,依照文献[2]中的实验设置,剔除掉带有"—"标签的数据,划分为训练集、验证集和测试集,各类别的分布情况如表 2 所示.类似地,把中文 CNLI 也划分为训练集、验证集和测试集,如表 3 所示.

Table 2 Training, Validation and Test Sets on SNLI

表 2 SNLI 上的训练集、验证集和测试集

Label	Training	Validation	Test
Entailment	183 416	3 329	3 368
Neutral	182 764	3 235	3 219
Contradiction	183 187	3 278	3 237
Total	549 367	9 842	9 824

Table 3 Training, Validation and Test Sets on CNLI

表 3 CNLI 上的训练集、验证集和测试集

Label	Training	Validation	Test
Entailment	29 738	3 485	3 475
Neutral	31 325	3 098	3 182
Contradiction	28 937	3 417	3 343
Total	90 000	10 000	10 000

使用 StanfordCoreNLP 工具包[①]对英文 SNLI 数据集进行 tokenization 处理,对中文 CNLI 数据集进行分词.实验中,使用预训练好的 300 维的英文 Glove 词向量[②],以及预训练好的 300 维的中文词向量[③].在训练的过程中,不进一步优化这些词向量.英文词表大小为 36 396,中文词表大小为 43 586.模型中的参数随机初始化为均值为 0、方差为 0.01 的正态分布.使用训练实例的真实标记与模型预测的分类结果之间的交叉熵(cross-entropy loss)作为代价函数.为了缓解训练中可能出现的过拟合问题,在模型的编码层使用了 dropout 技术[25].实验代码基于 TensorFlow1.13 实现.

基于验证集上的最优性能选择模型中超参(hyper parameters)的取值.实验中发现,表 4 中所列各超参的取值,既适用于英文 SNLI 数据集,也适用于中文 CNLI 数据集.某种程度上,这也反映了所提方法的稳定性.

Table 4 Values of Hyper Parameters

表 4 超参的值

Hyper Parameters	Values
Embedding Size d	300
Batch Size	128
Learning Rate	0.001
Window w	1
Maximum Time Step t	9
Dropout Rate	0.2
Maximum Length	50
Optimizer	Adam
Dimensions of Hidden Layers in F	[200,200]
Dimensions of Hidden Layers in G	[200,200]
Dimensions of Hidden Layers in MLP	[200,200]
Nonlinear Function	ReLU

4.2 结 果

为了验证所提方法在文本蕴含识别任务上的有效性,我们对比基准模型:

1) DeBiAttn_2016[18].直接把词向量作为交互层的输入,没有使用编码层.该模型首次把双向注意力机制(BiAttn)引入文本蕴含识别中,取得了较好的效果.

2) ESIM_2017[20].基于 Bi-LSTM 对前提和假设分别编码,交互层基于 BiAttn 建模局部语义表示,聚合层使用另一个 Bi-LSTM 计算全局语义关系,取得了当时最好的识别效果.

3) MwAN_2018[21].采用与 ESIM_2017 类似的模型,不同之处在于综合了 BiAttn 的 4 种不同计算方法建模局部语义关系.

4) KIM_2018[26].采用与 ESIM_2017 类似的模型,并集成了 WordNet 中的同义词对、反义词对、上下位词对等外部知识.

5) DAN_2018[27].联合英文文本蕴含识别和篇章连接词预测 2 个任务,可以看作是利用了额外训练数据的半监督方法.

6) MTDNN+BERT_2019[28].以大规模预训练的 $\text{BERT}_{\text{Large}}$ 模型[29]为基础,在多任务框架下联合训练多个语义理解任务,取得了当前最好的性能.

除了上述基准模型之外,实验中我们还对比了一些简化的模型,分别用于验证本文改进之处的效果:

1) Bi-LSTM+BiAttn 与 S-LSTM+BiAttn[④].在这 2 个基本的模型中编码层以 Bi-LSTM/S-LSTM 学习前提和假设的语义表示.

2) Ours-1.本文提出模型的一种简化,即把式(1)中的 $\boldsymbol{g}_a$ 和 $\boldsymbol{g}_b$ 去掉,用于验证编码层信息交换的作用.

3) Ours-2.本文提出模型的另一种简化,即把式(6)中与 g_b^{t-1} 相关的部分去掉,用于验证在交互层考虑全局语义表示 $\boldsymbol{g}_a$ 和 $\boldsymbol{g}_b$ 的作用.

4) Ours-3.本文提出的模型.

从表 5 中的实验结果可以看出:1)与没有利用外部资源的模型 DeBiAttn_2016, ESIM_2017, MwAN_2018 相比,我们提出的模型(Ours-3)取得了相同或更高的准确率.2)与使用了外部资源的 KIM_2018 和 DMAN_2018 相比,我们提出的模型取得了可比的性能.3)MTDNN+BERT_2019 取得了当前最好的性能,其性能的提高主要来源于基于超大规模语料预训练的 $\text{BERT}_{\text{Large}}$ 模型的使用.从某种角度来说,不能直接与本文所提出的模型进行对比.4)表 5 的下半部分中,S-LSTM+BiAttn 的效果略好于 Bi-LSTM+BiAttn 的效果,说明把 S-LSTM

① https://stanfordnlp.github.io/CoreNLP/

② http://nlp.stanford.edu/data/glove.840B.300d.zip

③ https://pan.baidu.com/s/1kwxiPouou6ecxyJdYmnkvw

④ S-LSTM+BiAttn 模型在交互层仅使用了短语级别的语义表示,没有利用句子级的语义信息.

用于文本蕴含识别任务的有效性;Ours-1 和 Ours-2 的效果都好于 S-LSTM+BiAttn 的效果,说明在编码层考虑前提和假设之间信息的交换和在交互层考虑句子级的全局信息这 2 处改进都是有效的;Ours-3 的效果好于 Ours-1 和 Ours-2 的效果,说明联合使用这 2 处改进能进一步提升识别的性能.综上所述,本文提出的模型是有效的,取得了同类模型中(没有使用外部资源)较好的识别性能.

Table 5 Results on SNLI

表 5 SNLI 上的实验结果

Model	Accuracy/%
DeBiAttn_2016[18]	86.3
ESIM_2017[20]	88.0
MwAN_2018[21]	88.3
KIM_2018[26]	88.6
DMAN_2018[27]	88.8
MTDNN+BERT_2019[28]	91.6
Bi-LSTM+BiAttn	87.0
S-LSTM+BiAttn	87.3
Ours-1	87.7
Ours-2	88.0
Ours-3	88.3

如表 6 所示,我们提出的模型在中文 CNLI 数据集上的实验结果与英文 CNLI 上的结果具有类似的趋势,再次验证了模型的有效性.

Table 6 Results on CNLI

表 6 CNLI 上的实验结果

Model	Accuracy/%
DeBiAttn_2016[18]	70.7
ESIM_2017[20]	72.2
Bi-LSTM+BiAttn	71.5
S-LSTM+BiAttn	72.0
Ours-1	72.3
Ours-2	72.6
Ours-3	73.2

4.3 超参分析

扩展的 S-LSTM 包括 2 个重要的超参:最大时刻 t 和窗口大小 w.具体地,最大时刻 t 表示建模句子的语义表示需要迭代的次数,其设定跟句子的长度无关.窗口大小 w 表明当前词与前后多少个词组成的上下文交换信息,可以理解为短语 ngram 的大小.式(6)中为了简洁把 w 设置为 1,计算得 $\boldsymbol{\xi}_i^t=(\boldsymbol{h}_{a_{i-1}}^{t-1},\boldsymbol{h}_{a_i}^{t-1},\boldsymbol{h}_{a_{i+1}}^{t-1})$,更一般地可表示为 $\boldsymbol{\xi}_i^t=(\boldsymbol{h}_{a_{i-w}}^{t-1},\boldsymbol{h}_{a_{i-1}}^{t-1},\cdots,\boldsymbol{h}_{a_i}^{t-1},\boldsymbol{h}_{a_{i+1}}^{t-1},\cdots,\boldsymbol{h}_{a_{i+w}}^{t-1})$.本节在 SNLI 验证集上探索了这 2 个超参对识别性能的影响.具体地,设置 $t\in\{1,3,5,7,9,11\}$ 和 $w\in\{1,2,3\}$,实验结果如图 3 所示.从图 3 中可以看出,$w=1$ 时的性能明显且稳定地好于 $w=2$ 和 $w=3$.这与我们的直觉是相符的,即 $w=1$ 时模型在不同时刻 t 建模的是大小为 3 的 ngram 及更大范围文本片段的语义,而 $w=2$ 时则是大小为 5 的 ngram 及更大范围文本片段的语义.对于最大时刻 t 来,当窗口设置较小时,则需要更大的 t 才能达到最佳效果,例如 $w=1$ 在 $t=9$ 时效果最佳.从图 3 中也可以看出,当窗口设置较小($w=1$ 或 $w=2$)时,不同的 t 上的效果是比较稳定的.

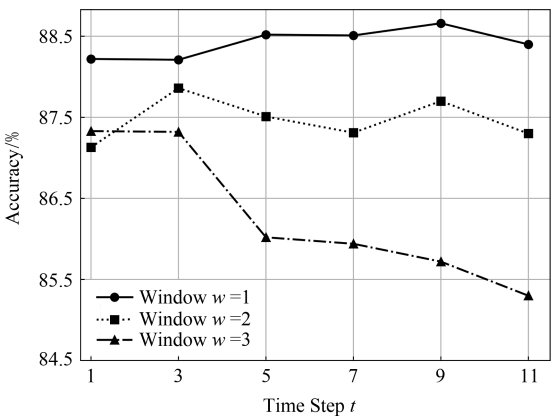

Fig. 3 Performance on the SNLI validation set

图 3 SNLI 验证集上的性能

4.4 错误分析

参照文献[30]分析 SNLI 中预测错误实例的方法,我们从 CNLI 的测试集中随机抽样了 100 个预测错误的实例进行分析.把导致预测错误的原因大致分成 6 类(示例如表 7 所示):

1) 词重复(30%).前提和假设中包含大量相同的词,会使模型倾向于预测其为蕴含关系.某种程度上,这说明现在的模型学到的很可能仅是浅层的特征,而不是真正的深层语义.

2) 否定词(6%).否定词的出现容易导致模型倾向于预测其为矛盾关系.例如,示例中的“不是”.

3) 反义词(9%).模型较难识别出前提和假设中具有相反意义的词.例如,示例中的“未煮过”和“熟”.

4) 背景知识(18%).没有一定的知识背景很难正确判别这些实例的关系.例如,示例中“乐队在舞台上”暗含“音乐会”的举行.

Table 7 Examples with Wrong Predictions in the CNLI Test Set

表 7 CNLI 测试集中预测错误的示例

Categories	Premise	Hypothesis	Gold	Predicted
Word Overlap	多尔克斯,你会告诉曼宁来这里跟我说话吗?	多尔克斯,请告诉曼宁去吃晚饭,然后过来跟我说话.	Neutral	Entailment
Negation	不管是不是疯了,他都认真地对待这个孵蛋的事情.	他非常认真地,实践了孵化鸡蛋的整个流程.	Entailment	Contradiction
Antonymy	一个小女孩正在吃未煮过的披萨配料.	一个女孩吃熟披萨.	Contradiction	Entailment
Real-World Knowledge	一位黑发女子在舞台上用相机拍下了一张乐队的照片.	一位女士在一场音乐会上.	Entailment	Neutral
Ambiguity	女孩正在展示.	女孩在显示器旁边.	Neutral	Entailment
Unknown	穿橙色背心的两个工人完成了他们的工作.	两名橙色猎人正在寻找鹿.	Contradiction	Neutral

5) 歧义(8%).有些实例,人类也很难作出正确的判断,存在一定的歧义.例如"在显示器旁边"有可能是"展示",也可能不是.

6) 其他(29%).有些实例没有明显的错误来源.

从以上错误分析可以看出,现有文本蕴含识别模型还有很多可以改进的地方.例如如何集成外部知识库以缓解错误 3),4);如何消除词重复带来的预测偏见(bias)等.

5 总 结

本文提出了一种基于扩展的 S-LSTM 的文本蕴含识别模型.具体地,从编码层中前提和假设的信息交换的建模、交互层中句子级别的全局语义的利用这 2 个方面对前人的工作进行改进.所提模型在英文 SNLI 和中文 CNLI 数据集上,都取得了同类方法中较好的识别性能.在未来的工作中,我们将探索把扩展的 S-LSTM 用于其他句子对相关的任务,如隐式篇章关系识别、复述识别等.

参 考 文 献

[1] Guo Maosheng, Zhang Yu, Liu Ting.Research advances and prospect of recognizing textual entailment and knowledge acquisition [J]. Chinese Journal of Computers, 2017, 40(4): 889-910 (in Chinese)
(郭茂盛, 张宇, 刘挺.文本蕴含关系识别与知识获取研究进展及展望[J]. 计算机学报, 2017, 40(4): 889-910)

[2] Bowman S R, Angeli G, Potts C, et al. A large annotated corpus for learning natural language inference [C] //Proc of the 2015 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2015: 632-642

[3] Hobbs J R, Stickel M, Martin P, et al. Interpretation as abduction [C] //Proc of the 26th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 1988: 95-103

[4] Ren Han, Sheng Yaqi, Feng Wenhe, et al. Recognizing textual entailment based on knowledge topic models [J]. Journal of Chinese Information Processing, 2015, 29(6): 119-126 (in Chinese)
(任函, 盛雅琦, 冯文贺, 等. 基于知识话题模型的文本蕴涵识别[J]. 中文信息学报, 2015, 29(6): 119-126)

[5] Sultan M A, Bethard S, Sumner T. Feature-rich two-stage logistic regression for monolingual alignment [C] //Proc of the 2015 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2015: 949-959

[6] Bos J, Markert K. Recognising textual entailment with logical inference [C] //Proc of the Conf on Human Language Technology and Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2005: 628-635

[7] Dozat T, Manning C D. Deep biaffine attention for neural dependency parsing [J]. arXiv preprint, arXiv:1611.01734, 2016

[8] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [J]. arXiv preprint, arXiv:1409.0473, 2014

[9] Cao Mingyu, Yang Zhihao, Luo Ling, et al. Joint drug entities and relations extraction based on neural networks [J]. Journal of Computer Research and Development, 2019, 56(7): 1432-1440 (in Chinese)
(曹明宇, 杨志豪, 罗凌, 等. 基于神经网络的药物实体与关系联合抽取[J]. 计算机研究与发展, 2019, 56(7): 1432-1440)

[10] Bowman S R, Gauthier J, Rastogi A, et al. A fast unified model for parsing and sentence understanding [C] //Proc of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2016: 1466-1477

[11] Liu Yang, Sun Chengjie, Lin Lei, et al. Learning natural language inference using bidirectional LSTM model and inner-attention [J]. arXiv preprint, arXiv:1605.09090, 2016

[12] Chen Qian, Zhu Xiaodan, Ling Zhenhua, et al. Recurrent neural network-based sentence encoder with gated attention for natural language inference [C] //Proc of the 2nd Workshop on Evaluating Vector Space Representations for NLP. Stroudsburg, PA: ACL, 2017: 36-40

[13] Mou Lili, Men Rui, Li Ge, et al. Natural language inference by tree-based convolution and heuristic matching [C] //Proc of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2016: 130-136

[14] Tan Yongmei, Liu Shuwen, Lü Xueqiang. CNN and BiLSTM based Chinese textual entailment recognition [J]. Journal of Chinese Information Processing, 2018, 32(7): 11-19 (in Chinese)
(谭咏梅, 刘姝雯, 吕学强. 基于 CNN 与双向 LSTM 的中文文本蕴含识别方法[J]. 中文信息学报, 2018, 32(7): 11-19)

[15] Shen Tao, Zhou Tianyi, Long Guodong, et al. Disan: Directional self-attention network for RNN/CNN-free language understanding [C] //Proc of the 32nd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2018: 5446-5455

[16] Conneau A, Kiela D, Schwenk H, et al. Supervised learning of universal sentence representations from natural language inference data [C] //Proc of the 2017 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2017: 670-680

[17] Jin Tianhua, Jiang Shan, Yu Dong, et al. Chinese chunked-based heterogeneous entailment parser and boundary Identification [J]. Journal of Chinese Information Processing 2019, 33(2): 17-25 (in Chinese)
(金天华, 姜珊, 于东, 等. 中文句法异构蕴含语块标注和边界识别研究[J]. 中文信息学报, 2019, 33(2): 17-25)

[18] Parikh A, Täckström O, Das D, et al. A decomposable attention model for natural language inference [C] //Proc of the 2016 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2016: 2249-2255

[19] Wang Zhiguo, Hamza W, Florian R. Bilateral multi-perspective matching for natural language sentences [J]. arXiv preprint, arXiv: 1702.03814, 2017

[20] Chen Qian, Zhu Xiaodan, Ling Zhenhua, et al. Enhanced LSTM for natural language inference [C] //Proc of the 55th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2017: 1657-1668

[21] Tan Chuanqi, Wei Furu, Wang Wenhui, et al. Multiway attention networks for modeling sentence pairs [C] //Proc of the 27th Int Joint Conf on Artificial Intelligence. Stockholm, Sweden: IJCAI, 2018: 4411-4417

[22] Gong Yichen, Luo Heng, Zhang Jian. Natural language inference over interaction space [J]. arXiv preprint, arXiv: 1709.04348, 2017

[23] Kim S, Kang I, Kwak N. Semantic sentence matching with densely-connected recurrent and co-attentive information [C] //Proc of the 33rd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2019: 6586-6593

[24] Zhang Yue, Liu Qi, Song Linfeng. Sentence-state LSTM for text representation [C] //Proc of the 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2018: 317-327

[25] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: A simple way to prevent neural networks from overfitting [J]. The Journal of Machine Learning esearch, 2014, 15(1): 1929-1958

[26] Chen Qian, Zhu Xiaodan, Ling Zhenhua, et al. Neural natural language inference models enhanced with external knowledge [C] //Proc of the 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2018: 2406-2417

[27] Pan Boyuan, Yang Yazheng, Zhao Zhou, et al. Discourse marker augmented network with reinforcement learning for natural language inference [C] //Proc of the 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2018: 989-999

[28] Liu Xiaodong, He Pengcheng, Chen Weizhu, et al. Multi-task deep neural networks for natural language understanding [C] //Proc of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 4487-4496

[29] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding [C] //Proc of the 2019 Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2019: 4171-4186

[30] Naik A, Ravichander A, Sadeh N, et al. Stress test evaluation for natural language inference [C] //Proc of the 27th Int Conf on Computational Linguistics. Stroudsburg, PA: ACL, 2018: 2340-2353

Hu Chaowen, born in 1993. Master candidate. Student member of CCF. His main research interests include deep learning and nature language processing.

Wu Changxing, born in 1981. PhD, lecturer. Member of CCF. His main research interests include nature language processing and machine learning.

Yang Yalian, born in 1982. Bachelor, lecturer. Her main research interests include English language literature and bilingual contrastive linguistics. (yalianyang@163.com)