文章编号：1003-0077（2011）00-0000-00

问答中的问句意图识别和约束条件分析*

孙鑫[1]，王厚峰[1]

（1. 北京大学计算语言学教育部重点实验室，北京市 100871）

摘要：意图识别（Intent Determination, ID）和约束条件分析（Slot Filling, SF）是口语理解（SLU）中的两个重要过程。前者是分类问题，判断话语意图；后者可以看作序列标注问题，给关键信息标特定标签。本文提出了一种 LSTM（Long Short-Term Memory Network）联合模型，同时结合了 CRF（Conditional Random Field）和注意力机制（Attention Model）。在 ID 问题上，将所有词语输出层向量的加权和用于分类；在 SF 问题上，考虑标签之间的转移，计算标签序列在全局的可能性。在中文数据集和 ATIS 英文数据集上的实验验证了本文所提方法的有效性。

关键词：长短期记忆网络；条件随机场；注意力机制

中图分类号：TP391　　**文献标识码：**A

Intent Determination and Slot Filling in Question Answering

Xin Sun[1], Houfeng Wang[1]

(1. MOE Key Lab of Computational Linguistics, Peking University, Beijing　100871, China)

Abstract: Intent determination (ID) and slot filling (SF) are two major tasks in spoken language understanding (SLU). The former is a classification problem, which judges the intention of utterance. The later can be treated as a sequence labeling problem, labelling key information as specific symbols. This paper proposed a LSTM (Long Short-Term Memory Network) joint model combined with attention and CRF (Conditional Random Field). In ID problem, the weighted sum of output layer's vectors was used in classification task as the utterance's representation. In SF problem, this paper considered the transfers between labels and computed the probabilities on the sequence-level. This model was verified in Chinese and English corpuses.

Key words: Long Short-Term Memory Network; Conditional Random Field; Attention Model

1 引言

口语对话系统、语音助手、自动客服等是当前自然语言处理研究的热点。这些应用的成功不仅取决于语音内容的识别，更在于对句子含义的理解。和自动语音识别（Automatic Speech Recognizer, ASR）是将说话人的语音转化为文字序列不同，口语理解（Spoken Language Understanding, SLU）侧重于确定口语问句包含的意图[1]，并提取出相应的约束条件，再交给对话或任务管理器（Dialog or Task Manager, DM）[2]，就可以完成用户的特殊需求。ASR、SLU 和 DM 顺序连接构成了典型的面向实际目标的会话理解系统[2]，是自动客户服务等应用的关键构成部分。本文研究 SLU 中的意图识别和约束条件分析，为口语理解提供帮助。

意图识别（Intent Determination, ID）旨在确定一句话的意图，可以看成分类问题。事先在该领域定义各种可能的意图类别，再用分类方法将问句分到某类之中。约束条件分析（Slot Filling, SF）则是提取出达成意图的关键信息，可以看成序列标注问题，即，将关键词标注为特定标签，其它词被标注为普通标签。

在表 1 中，我们采用 In/Out/Begin(IOB)标签来标注“西安到拉萨 8 月 1 号到 5 号有没有打折的飞机票”这句话。“西安”被标注为起点，“拉萨”为终点，“8 月 1 号到 5 号”被标

*收稿日期：　定稿日期：

基金项目：国家自然科学基金（编号：No.61370117, No.61433015）

注为出发时间，“打折”是票价要求。这句话的意图是“查机票”。

表 1：ID 和 SF 问题样例

领域	航班
意图	查机票
句子	**标签**
西安	B-起点
到	O
拉萨	B-终点
8 月	B-出发时间
1 号	I-出发时间
到	I-出发时间
5 号	I-出发时间
有没有	O
打折	B-票价要求
的	O
飞机票	O

意图和约束条件这两个问题通常被当作独立的两个问题处理，前者可以用各种机器学习的分类方法，如支持向量机（Support Vector Machine, SVM）、逻辑回归等解决；后者可以用序列标注模型实现，如条件随机场（Conditional Random Field, CRF）、循环神经网络（Recurrent Neural Networks, RNN）等。但这两个问题是相关的，两种信息可以相互帮助、相互影响[3]，例如“查机票”类型的问句，很可能出现“起点”“终点”“时间”这类标签；反过来，这些标签的出现也很可能意味着问句的意图是“查机票”。目前已有研究采用联合模型，同时识别意图和提取出关键信息。

2 相关工作

单一模型将意图识别和约束条件分析作为两个独立的问题。

意图识别是分类问题。可以使用分类方法求解，如 SVM 模型[4]，Adaboost 方法[5]等。这类传统分类方法的一般步骤是，提取特征作为问句的输入，再用训练好的分类模型进行多分类。缺点是需要人工设定特征。当数据集发生变化时，需要重新设计特征。这往往会演变成特征设计、特征选取等问题，失去了对课题本质的关注。

约束条件分析一般被视为序列标注问题。条件随机场[6]是解决这类问题的经典方法。这类方法一般步骤是，将训练数据中的被标注序列和标签看作很多节点，设计两个或多个节点之间的关系作为特征，训练出这些特征的权重。最后在测试数据上，将更能反映这些特征的标签序列作为结果。缺点同样是需要人工设定特征模板。

深度学习方法可以避免人工构建特征，例如，RNN 方法[7]及各种改进模型如 LSTM、GRU 等。这类方法在计算每个词标签的概率时，是局部进行归一的，而局部归一容易出现偏置[8]。CRF 方法是全局归一，关注整个标签序列占所有可能标签序列的概率。结合 CRF 的 RNN 模型[8]可以提高约束条件分析的效果。不过，这种方法只关注单个问题，没有解决意图识别问题，忽略了意图识别模块带来的额外信息。

论文[3]提出了基于 RNN 的联合模型，以 GRU 作为基本单元，同时借鉴 CRF 的思想，在目标函数中加入了标签状态转移的评估。缺点是在解决意图识别问题时，只是取样了 GRU 每次输出的最大值，有信息损失。其次虽然加入了标签之间的转移评估，但是仍然没有在全局考虑序列的概率。论文[9]在处理意图识别问题时，使用了注意力机制，能较好地区分哪些

词对分类起到了更大的作用，缺点仍然是约束条件分析时，简单地使用 softmax 计算概率，没有考虑整体序列的概率分布，有标注偏置的隐患。

综上所述，当前 RNN 联合模型没有很好地兼顾意图识别和约束条件分析两个方面，前者需要给有用的词汇更大的权重，提取更有用的信息，简单取最大值的操作会损失信息；后者需要在整体标签序列上计算概率，可以避免标注偏置问题。

3 模型

3．1 常规处理

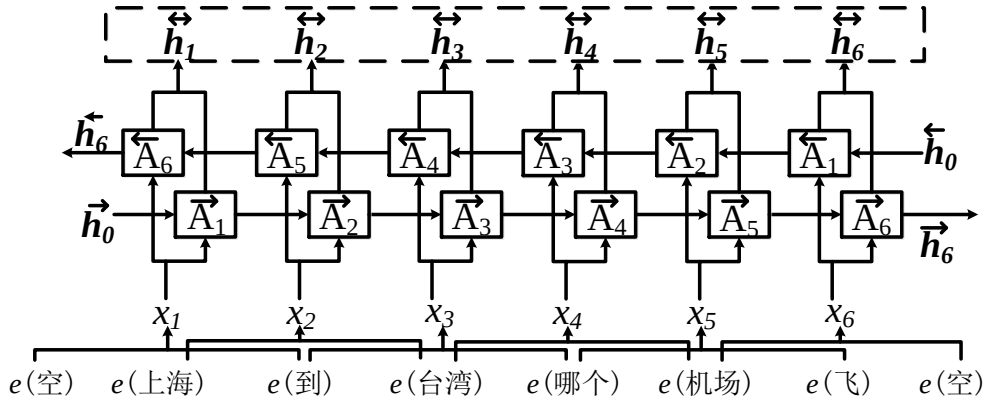

图 1：常规处理模型

基于 LSTM 的联合模型的基本框架如图 1 所示。设置词向量矩阵$\boldsymbol{E_{emb}} \in \mathbb{R}^{(|V|+1)*|e|}$，$|e|$表示词向量维数，$|V|$表示总词数，增加一行作为取上下文窗口时空白词的词向量。采用词向量和上下文窗口，将当前词与窗口内的词的词向量连接，作为当前$\boldsymbol{x_t}$，输入到 LSTM 单元，设置取样前后各 d 个词汇，如公式（1）所示：

$$\boldsymbol{x_t} = [e(w_{t-d}), \dots, e(w_t), \dots, e(w_{t+d})] \quad (1)$$

$e(w_t)$表示当前词w_t的词向量，相当于在词向量表$\boldsymbol{E_{emb}}$中查询到这个词对应的那一行向量。向量$e(w_t) \in \mathbb{R}^{|e|}$，向量$\boldsymbol{x_t} \in \mathbb{R}^{(2d+1)|e|}$，“[]”表式向量的简单拼接。将$\boldsymbol{x_t}$输入到正反 LSTM 模块中，正反两个模块分别使用各自的一套参数。参数$\overrightarrow{LSTM} = \{\overrightarrow{\boldsymbol{W_f}}, \overrightarrow{\boldsymbol{W_\iota}}, \overrightarrow{\boldsymbol{W_c}}, \overrightarrow{\boldsymbol{W_\iota}}, \overrightarrow{\boldsymbol{U_f}}, \overrightarrow{\boldsymbol{U_\iota}}, \overrightarrow{\boldsymbol{U_c}}, \overrightarrow{\boldsymbol{U_\iota}}, \overrightarrow{\boldsymbol{b_f}}, \overrightarrow{\boldsymbol{b_\iota}}, \overrightarrow{\boldsymbol{b_c}}, \overrightarrow{\boldsymbol{b_\iota}}, \overrightarrow{\boldsymbol{h_0}}, \overrightarrow{\boldsymbol{c_0}}\}$,参数$\overleftarrow{LSTM} = \{\overleftarrow{\boldsymbol{W_f}}, \overleftarrow{\boldsymbol{W_\iota}}, \overleftarrow{\boldsymbol{W_c}}, \overleftarrow{\boldsymbol{W_\iota}}, \overleftarrow{\boldsymbol{U_f}}, \overleftarrow{\boldsymbol{U_\iota}}, \overleftarrow{\boldsymbol{U_c}}, \overleftarrow{\boldsymbol{U_\iota}}, \overleftarrow{\boldsymbol{b_f}}, \overleftarrow{\boldsymbol{b_\iota}}, \overleftarrow{\boldsymbol{b_c}}, \overleftarrow{\boldsymbol{b_\iota}}, \overleftarrow{\boldsymbol{h_0}}, \overleftarrow{\boldsymbol{c_0}}\}$。正向 LSTM 的公式[10]如(2)-(7)所示：

$$\overrightarrow{\boldsymbol{f_t}} = sigmoid\ (\overrightarrow{\boldsymbol{W_f}}\boldsymbol{x_t} + \overrightarrow{\boldsymbol{U_f}}\overrightarrow{\boldsymbol{h_{t-1}}} + \overrightarrow{\boldsymbol{b_f}}) \quad (2)$$

$$\overrightarrow{\boldsymbol{\iota_t}} = sigmoid\ (\overrightarrow{\boldsymbol{W_\iota}}\boldsymbol{x_t} + \overrightarrow{\boldsymbol{U_\iota}}\overrightarrow{\boldsymbol{h_{t-1}}} + \overrightarrow{\boldsymbol{b_\iota}}) \quad (3)$$

$$\overrightarrow{\boldsymbol{c_t'}} = tanh(\overrightarrow{\boldsymbol{W_c}}\boldsymbol{x_t} + \overrightarrow{\boldsymbol{U_c}}\overrightarrow{\boldsymbol{h_{t-1}}} + \overrightarrow{\boldsymbol{b_c}}) \quad (4)$$

$$\overrightarrow{\boldsymbol{c_t}} = \overrightarrow{\boldsymbol{c_{t-1}}} * \overrightarrow{\boldsymbol{f_t}} + \overrightarrow{\boldsymbol{c_t'}} * \overrightarrow{\boldsymbol{\iota_t}} \quad (5)$$

$$\overrightarrow{\boldsymbol{o_t}} = sigmoid\ (\overrightarrow{\boldsymbol{W_o}}\boldsymbol{x_t} + \overrightarrow{\boldsymbol{U_o}}\overrightarrow{\boldsymbol{h_{t-1}}} + \overrightarrow{\boldsymbol{b_o}}) \quad (6)$$

$$\overrightarrow{\boldsymbol{h_t}} = \overrightarrow{\boldsymbol{o_t}} * tanh(\overrightarrow{\boldsymbol{c_t}}) \quad (7)$$

LSTM 模块在 RNN 的基础上增加了记忆信息$\overrightarrow{\boldsymbol{c_t}}$这个向量，在每个时刻通过门的处理遗忘部分信息，再增加部分信息，最后过滤得到这个时刻的隐藏层输出$\overrightarrow{\boldsymbol{h_t}}$，“$*$”表示按元素相乘。所有权重 $\boldsymbol{W} \in \mathbb{R}^{|h|*(2d+1)|e|}$，所有权重 $\boldsymbol{U} \in \mathbb{R}^{|h|*|h|}$，所有向量参数 $\boldsymbol{b}$,$\boldsymbol{h}$,$\boldsymbol{c} \in \mathbb{R}^{|h|}$。其中内部计算出的$\boldsymbol{f_t}$,$\boldsymbol{\iota_t}$,$\boldsymbol{c_t'}$,$\boldsymbol{c_t}$,$\boldsymbol{o_t}$,$\boldsymbol{h_t} \in \mathbb{R}^{|h|}$。$|h|$表示隐藏层维数，$d$ 表示上下文窗口大小，$|e|$表示词向量维数。反向结构与正向结构计算方法相同，将最终结果拼接使用。如公式（8）所示：

$$\overleftrightarrow{\boldsymbol{h_t}} = [\overrightarrow{\boldsymbol{h_t}}, \overleftarrow{\boldsymbol{h_t}}] \quad (8)$$

该模块需要训练的参数是$\theta = \{\boldsymbol{E_{emb}}, \overrightarrow{LSTM}, \overleftarrow{LSTM}\}$。

本文引入 attention 实现意图分析，通过增加 CRF 识别约束条件。

3．2 CRF 机制

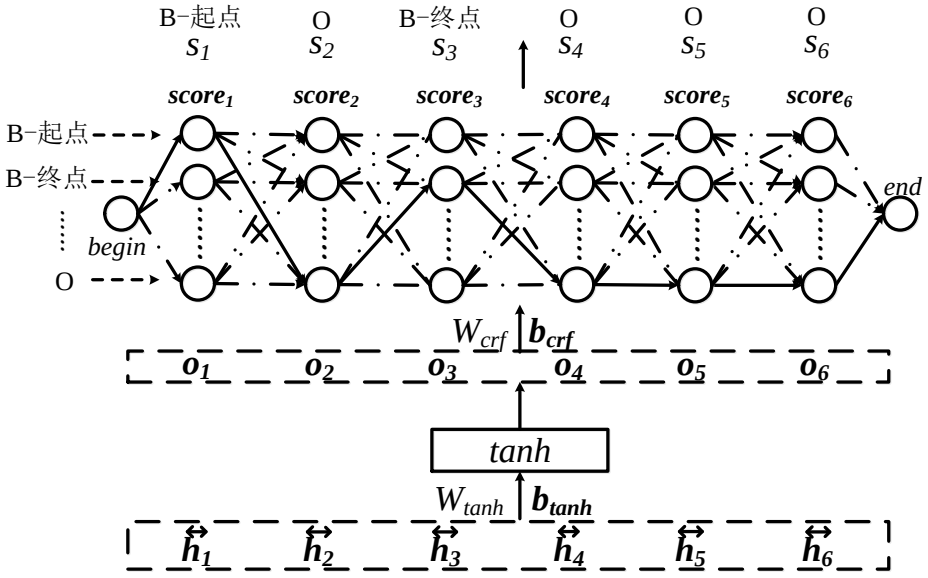

图 2：crf 机制

图 2 所示的框架不再和其他模型一样直接用 softmax 函数计算每个标签的概率，而是借鉴链式 CRF[6]的方法考虑标签之间的转移和整体上标签序列的概率。从单个时刻看，当前词对应的标签都有相应分数，分数越大，对应标签越有可能；从标签序列的整体来看，前后标签有转移分数，分数越高，越可能出现这种标签的转移。将这两个分数相加，分数最高的标签序列就是预测结果。先对隐藏层输出做预处理，如公式（9）所示：

$$\boldsymbol{o_t} = tanh(\boldsymbol{W_{tanh}}\overleftrightarrow{\boldsymbol{h_t}} + \boldsymbol{b_{tanh}}) \tag{9}$$

双向 LSTM 隐藏层输出拼接得到$\overleftrightarrow{\boldsymbol{h_t}} \in \mathbb{R}^{2|h|}$，它对应的权重$\boldsymbol{W_{tanh}} \in \mathbb{R}^{|h|*2|h|}$，参数向量$\boldsymbol{b_{tanh}} \in \mathbb{R}^{|h|}$，$\boldsymbol{o_t} \in \mathbb{R}^{|h|}$，它所有元素的取值在[-1,1]之间。$|h|$仍表示隐藏层维数。接下来，求解这个时刻的词对应的所有标签的分数，如公式（10）所示：

$$\boldsymbol{scores_t} = \boldsymbol{W_{crf}}\boldsymbol{o_t} + \boldsymbol{b_{crf}} \tag{10}$$

其中$\boldsymbol{W_{crf}} \in \mathbb{R}^{|S|*|h|}$,$\boldsymbol{scores_t}$,$\boldsymbol{b_{crf}} \in \mathbb{R}^{|S|}$，$|S|$表示标签个数。向量$\boldsymbol{scores_t}$每个元素表示对应标签的分数，分数越大越有可能。同时设置转移矩阵$\boldsymbol{E_{trans}} \in \mathbb{R}^{(|S|+2)*(|S|+2)}$，表示标签之间的转移分数，在$|S|$个标签的基础上增加了“开始”和“结束”两个标签。一个标签序列的总分数是由每个标签的分数和它们之间的转移来决定的，如公式（11）所示：

$$\begin{aligned} & score_{total}(\boldsymbol{s_{1:T}}) \\ =\ & transfer\ score\ of\ \boldsymbol{s_{1:T}} + score\ of \boldsymbol{s_{1:T}} \\ =\ & \sum_{t=1}^{T+1}(\boldsymbol{E_{trans}}(\boldsymbol{s_{t-1}}, \boldsymbol{s_t}) + \boldsymbol{scores_t}(\boldsymbol{s_t})) \end{aligned} \tag{11}$$

$\boldsymbol{s_{1:T}}$表示标签序列，时刻总个数 T 等于$|L|$，也就是该词语序列的长度。$score_{total}(\boldsymbol{s_{1:T}})$是对该标签序列的总分数的衡量。$\boldsymbol{E_{trans}}(\boldsymbol{s_{t-1}}, \boldsymbol{s_t})$表示转移矩阵$\boldsymbol{E_{trans}}$第$\boldsymbol{s_{t-1}}$行，第$\boldsymbol{s_t}$列的值，即 *t-1* 时刻的标签到 *t* 时刻标签的转移分数。$\boldsymbol{scores_t}(\boldsymbol{s_t})$表示$\boldsymbol{scores_t}$的第$\boldsymbol{s_t}$个元素，即当前标签是$\boldsymbol{s_t}$时的分数。$\boldsymbol{s_0}$和$\boldsymbol{s_{T+1}}$分别为“开始”和“结束”标签，它们对应的$\boldsymbol{scores_t}(\boldsymbol{s_0})$和$\boldsymbol{scores_t}(\boldsymbol{s_{T+1}})$取零。为了在标签序列的整体进行概率归一，本文利用 softmax 的思想，对总分数进行概率转换，如公式（12）所示：

$$
\begin{aligned}
& s(\boldsymbol{s}_{1:T}, \theta) \\
&= log\big(P(\boldsymbol{s}_{1:T}|L)\big) \\
&= log\left(\frac{exp\big(score_{total}(\boldsymbol{s}_{1:T})\big)}{\sum_{all\ possibles_{1:T}} exp\big(score_{total}(\boldsymbol{s}_{1:T})\big)}\right) \\
&= log\left(\frac{exp(\sum_{t=1}^{T+1}(\boldsymbol{E}_{trans}(\boldsymbol{s}_{t-1},\ \boldsymbol{s}_t)\ +\ \boldsymbol{scores}_t(\boldsymbol{s}_t)))}{Z_{CRF}}\right) \\
&= log\left(\frac{exp(\sum_{t=1}^{T+1}(\boldsymbol{E}_{trans}(\boldsymbol{s}_{t-1},\ \boldsymbol{s}_t)\ +\ \boldsymbol{scores}_t(\boldsymbol{s}_t)))}{\sum_{\boldsymbol{s}}(exp(\sum_{t=1}^{T+1}(\boldsymbol{E}_{trans}(\boldsymbol{s}_{t-1},\ \boldsymbol{s}_t)\ +\ \boldsymbol{scores}_t(\boldsymbol{s}_t))))}\right) \\
&= log(exp\big(score_{total}(\boldsymbol{s}_{1:T})\big)) - log(Z_{CRF})
\end{aligned}
\tag{12}
$$

Z_{CRF}是归一化因子，即，分子序列表示的所有可能取值之和。L 表示原序列。该公式的含义是在参数θ的情况下，求标签序列$\boldsymbol{s}_{1:T}$的分数占所有可能标签分数的比例。取概率最大的标签序列$\boldsymbol{s}_{1:T}$作为预测$\widehat{\boldsymbol{s}_{1:T}}$，如公式（13）所示：

$$
\widehat{\boldsymbol{s}_{1:T}} = argmax_{\boldsymbol{s}_{1:T}} s(\boldsymbol{s}_{1:T}, \theta) \tag{13}
$$

训练时，约束条件分析部分的损失函数如公式（14）所示：

$$
\begin{aligned}
& L_{slot}(\tilde{\boldsymbol{s}}, \theta) \\
&= -s(\tilde{\boldsymbol{s}}, \theta) \\
&= log\left(\sum_{all\ possible\ \boldsymbol{s}_{1:T}} exp\big(score_{total}(\boldsymbol{s}_{1:T})\big)\right) \\
&\quad - log\big(exp\big(score_{total}(\tilde{\boldsymbol{s}}_{1:T})\big)\big)
\end{aligned}
\tag{14}
$$

即对正确标注该序列的概率的对数值取反，化简之后是两个对数值的差值。该模块增加了参数$\boldsymbol{E}_{trans}, \boldsymbol{W}_{tanh}, \boldsymbol{b}_{tanh}, \boldsymbol{W}_{crf}, \boldsymbol{b}_{crf}$。

3．3 ATTENTION

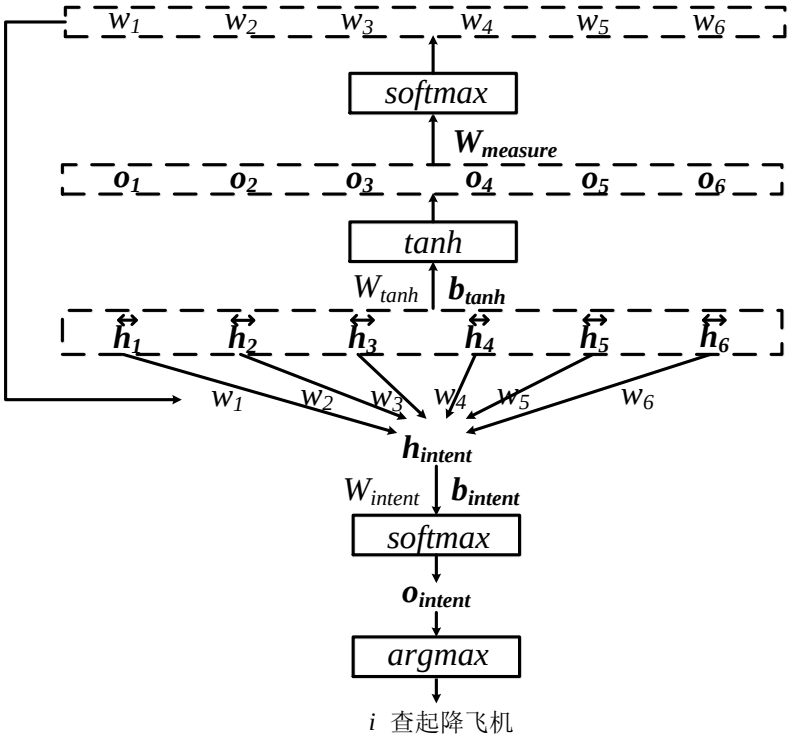

图 3：attention 机制

如图 3 所示，attention 方法[11]是将这些$\overrightarrow{\boldsymbol{h}_t}$的加权和作为整个句子的表示向量，所以关键

在于求解每个时刻对应的权重。第一步和 CRF 模块相同，经过一个相同的 tanh 层，得到$\boldsymbol{o_t}$，如公式（9）所示。建立一个向量$\boldsymbol{W_{measure}} \in \mathbb{R}^{|h|}$，表示对信息的衡量，$|h|$表示隐藏层维数。根据公式（15）得到当前词信息$\overleftrightarrow{\boldsymbol{h_t}}$对应的权重$w_t$：

$$w_t = \frac{exp(\boldsymbol{W_{measure}^T o_t})}{\sum_{i=1}^{|L|} exp(\boldsymbol{W_{measure}^T o_i})} \tag{15}$$

$|L|$表示词序列的长度。公式（15）的含义就是所有时刻的输出向量$\boldsymbol{o_t}$都要和$\boldsymbol{W_{measure}}$的转置相乘，再用指数函数处理后得到$|L|$个标量，每个时刻的权重就是对应标量占所有标量之和的比例。接着将每个时刻的隐藏层输出$\overleftrightarrow{\boldsymbol{h_t}}$加权相加，得到整个句子的向量表示$\boldsymbol{h_{intent}} \in \mathbb{R}^{2|h|}$，如公式（16）所示：

$$\boldsymbol{h_{intent}} = \sum_{i=1}^{T} w_t \overleftrightarrow{\boldsymbol{h_t}} \tag{16}$$

接下来对这个向量表示$\boldsymbol{h_{intent}}$进行处理，如公式（17）（18）所示：

$$\boldsymbol{o_{intent}} = softmax(\boldsymbol{W_{intent} h_{intent}} + \boldsymbol{b_{intent}}) \tag{17}$$

$$\hat{\imath} = argmax_i(\boldsymbol{o_{intent}}(i)) \tag{18}$$

$\boldsymbol{o_{intent}}$的每个元素代表对应意图类别的概率，取其中概率最大的类别作为预测类别。$\boldsymbol{W_{intent}} \in \mathbb{R}^{|I|*|h|}$，$\boldsymbol{o_{intent}}$，$\boldsymbol{b_{intent}} \in \mathbb{R}^{|I|}$，$|I|$表示意图类别个数，$\boldsymbol{o_{intent}}(i)$表示向量$\boldsymbol{o_{intent}}$的第 *i* 个元素，即该句意图为 *i* 的预测概率。意图识别模块的损失函数如公式（19）所示：

$$L_{intent}(\tilde{\imath}, \theta) = -log(\boldsymbol{o_{intent}}(\tilde{\imath})) \tag{19}$$

$\tilde{\imath}$代表这句话的正确意图，$\boldsymbol{o_{intent}}(\tilde{\imath})$表示预测结果中意图正确的概率。所以公式（19）就是在当前参数是θ的情况下，将正确意图的概率取对数，再取反作为损失函数。

CRF-attention-LSTM 模型训练的目标函数也是把所有句子的损失相加，如公式（20）所示：

$$L(\theta) = \sum_{\tilde{\boldsymbol{s}}_{\boldsymbol{i}}, \tilde{\imath} \in \mathbb{D}} \alpha L_{slot}(\tilde{\boldsymbol{s}}, \theta) + L_{intent}(\tilde{\imath}, \theta) \tag{20}$$

$\mathbb{D}$表示训练集集合，α表示 SF 模块损失的权重，可以调节两个模块的重要性。$\tilde{\boldsymbol{s}}_{\boldsymbol{i}}, \tilde{\imath} \in \mathbb{D}$表示遍历该训练集中所有句子的真实标签序列和意图。公式（20）就是在当前参数为θ的情况下，将所有训练集句子的损失按权重求和。该模块的参数是$\boldsymbol{W_{measure}}$。

4 实验

4．1 数据集

本文在两个数据集上进行实验，分别是中文数据集和英文数据集。

表 2：中文数据

领域	问句数量	意图数量	意图举例	约束条件类型数量	约束条件类型举例
航班	1117	13	查机票、问票价、问航班……	12	航班号、起点、终点、出发时间……
天气	1134	16	问天气、问温度、问空气质量……	2	目的地、时间
快递	635	16	查起点、查终点、查状态……	6	快递公司、起点、终点、重量\体积、转单公司

中文数据集的相关情况如表 2 所示，是从“百度知道”中爬取后加工得到的，包括 3 个领域，即，航班、天气和快递相关的问题。

表 3：英文数据

名称	训练集数量	测试集数量	意图数量	意图类型举例	约束条件类型数量	约束条件类型举例
ATIS	4978	893	18	问距离、问餐食、问航班号……	63	到达时间、花费、航班号……

英文数据集是口语理解问题中最常用的 ATIS 数据集，如表 3 所示。该数据集还包含词语的类型信息。有研究[12]认为，类型信息在 SLU 问题中是不常见的，所以本文不使用这种额外的信息。

4．2 评价标准

本文使用准确率来评估意图识别。在 ATIS 数据集中，有些数据可能有不止一个意图，本文按照论文[2]的方法，只要预测意图在正确意图之列就算作预测正确。

实验使用了 F1-分数来衡量约束条件分析结果，一个约束条件提取正确表示它的范围和类型都是正确的。F1-分数由通用的 CoNLL 评价脚本[1]获得。

4．3 对比方法

4．3．1 中文数据集中的 baseline

SVM 用于意图识别，提取的特征主要是一系列关键字是否存在的二元特征；CRF 方法用于约束条件分析。另外，本文也同几种不同的 RNN 模型比较了实验效果，其中，RNN-ID 和 RNN-SF 分别为普通 RNN 模型单独解决这两个问题时的方法，RNN-joint 为普通 RNN 联合模型，bi-RNN-joint 为双向 RNN 联合模型，bi-LSTM 为双向 LSTM 联合模型，它们在 SF 问题上直接用 softmax 处理取最大概率的标签，在 ID 问题上用最大池取样作为整句的代表向量。CRF-attention-LSTM 为本文方法，结合了 CRF 和 attention 的双向 LSTM 联合模型。

4．3．2 英文数据集中的 baseline

SVM：论文[6]使用前后向移动序列化 SVM 分类器，用于约束条件分析；

CRF：论文[7]提供的 baseline；

RNN：论文[7]提供的 RNN 约束条件分析方法；

Boosting：论文[2]使用了 AdaBoost.MH 方法处理意图识别方法；

Sentence simplification：论文[13]使用 AdaBoost.MH 方法处理意图识别，用 CRF 处理 SF；

RecNN：论文[14]使用的递归神经网络联合模型，后加入了 Viterbi 算法优化标签序列整体的评估，模型使用了额外的语义信息；

GRU-joint：论文[3]使用的 GRU 联合模型，在意图识别部分使用最大池，在约束条件分析部分考虑了标签之间的转移。

4．4 训练细节

中文数据集的实验中，本文采用 10-折交叉验证，不使用字标注，而是将句子分词后处理；英文数据集，使用划分好的训练集和测试集。模型的词向量维数均设置为 100，隐藏层维数为 100，上下文窗口大小设置为 1。中文数据集使用梯度下降方法更新参数，初始学习率设计为 0.0627；英文数据集使用 AdaDelta 方法[15]更新参数。实验结果表明，分别使用这两种参数更新方式，效果更好。代码使用 theano 进行编写。

[1] http://www.cnts.ua.ac.be/conll2000/chunking/output.html

4．5 实验结果与分析

4．5．1 中文数据集实验结果与分析

表 4：中文语料实验结果

领域	航班		天气		快递	
方法\任务	意图识别	约束条件分析	意图识别	约束条件分析	意图识别	约束条件分析
svm	87.9141		92.6808		87.7165	
crf		82.7179		86.1443		80.7815
RNN-ID	94.718		94.3563		87.4016	
RNN-SF		83.4469		88.179		85.0853
RNN-joint	92.6589	82.6842	95.1499	87.8971	87.4016	83.2895
bi-RNN-joint	93.8227	83.2459	95.0617	87.2659	**89.7638**	84.1105
bi-LSTM	95.7923	85.4826	95.5026	88.8738	89.4488	**85.892**
CRF-attention-LSTM	**95.7923**	**85.885**	**95.9436**	**89.5886**	88.0315	85.8123

实验结果如表 4 所示，在中文数据集上，先采用了经典的分类方法 SVM 和序列标注方法 CRF。最简单的 RNN 模型分别独立处理意图识别和约束条件分析，均超过了 SVM 和 CRF 的效果，只是数据量较少的“快递”领域在意图识别上稍有下降。RNN 能捕获长距离的依赖关系，在数据量充足时更加明显。

RNN 联合模型相比较 RNN 单一模型，效果有所下降，只是在“天气”领域的意图识别任务上有提高。双向的 RNN 联合模型在普通 RNN 联合模型基础上，有了一定的提升，说明后向的信息包括了一些有用的内容，只是在“天气”领域效果略有下降。这可能是因为天气领域的标签最少而类别最多，它的局限不在于前后向的信息，而在于模型本身处理、记忆信息的能力。双向的 LSTM 联合模型基本提高了各个领域的效果，因为 LSTM 相较普通 RNN 更能甄别、保留有用信息，分辨能力更强大。

最终的 CRF-attention-LSTM 模型，基本达到了最好的结果，只是在第三个领域稍低于之前的方法，因为第三个领域的数据量较少，不同标签的数量分布不均，之前的简单模型更适合它的复杂度，即便如此，最终模型在约束条件识别上的效果也与它们基本持平。

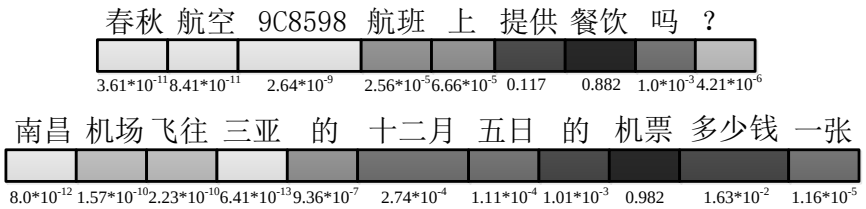

图 4：attention 结果示例

attention 模块的效果示例如图 4 所示，在这两个例子中，模型能够较好地聚焦在关键词

上，颜色越深、数值越大说明这个词越重要。

在中文数据集的实验表明，RNN 模型能够较好地解决意图识别和约束条件分析问题；当改为双向、LSTM 模块之后，效果更佳；而引入 CRF 和 attention 的 LSTM 联合模型，能够在两个问题上更好地利用信息进行分类和标注，达到最优的效果。

4．5．2 英文数据集实验结果与分析

表 5：英文语料实验结果

数据集	ATIS	
方法\任务	**意图识别**	**约束条件分析**
SVM[6]		89.76
CRF[7]		92.94
RNN[7]		95.06
Boosting[2]	95.50	
Sentence simplification[13]	96.98	95.00
RecNN[14]	95.40	93.22
RecNN+Viterbi[14]	95.40	93.96
GRU-joint[3]	**98.10**	95.49
本文 CRF-attention-LSTM	97.9843	**95.7**

为了验证所提模型 CRF-attention-LSTM 的有效性，本文使用英文数据集 ATIS 进行了实验。有些工作使用了 ATIS 中的额外信息，比如词的类型信息、语义信息等等，本文挑选了一些没有使用词的类型信息的相关结果作为比较，结果如表 5 所示。

对于单一模型而言，CRF 能比 SVM 更好地处理序列标注问题，RNN 因为更能捕获长距离依赖，效果更好。Sentence simplification 不是严格意义上的联合模型，它使用了不同的分类器处理两个问题，取得了较好的结果。RecNN 结合了语法信息和深度学习，但不适合这个数据集，效果不佳。本文提出的 CRF-attention-LSTM 在约束条件的分析上好于各种 baseline，在意图识别上略低于 GRU 的联合模型。这表明，本文所提方法在解决意图识别和约束条件分析上是有效的。

5 总结与展望

本文提出了结合了 CRF 和 attention 机制的 LSTM 联合模型，在意图识别部分通过 attention 提取和筛选有用的词语用于分类任务，在约束条件分析部分利用 CRF 方法，对标签序列整体上进行归一化。通过在中文和英文数据集上进行实验比较，验证了方法的有效性。结合 CRF 能提升模型整体评估标签序列的能力，attention 可以提高信息筛选能力，在词语中选择更重要的词汇用于分类。

本文将中文数据集分为三个领域进行实验，主要是考虑到了 SLU 任务中本身就有区分领域这一模块，意图识别和约束条件分析就是在领域确定的情况下进行处理的。也可以合在一起进行实验，这样数据量充足，且便于比较，只是和现实中 SLU 任务的真实情况不太相符。在后续工作中，可以加入 ATIS 数据集中的词语类型信息，和更多利用了这些额外信息的模型进行对比，以更好地验证模型效果。

参考文献

[1]. Wang Y Y, Deng L, Acero A. Spoken language understanding[J]. IEEE Signal Processing Magazine, 2005, 22(5):16-31.

[2]. Tur G, Hakkani-Tür D, Heck L. What is left to be understood in ATIS?[C]// Spoken Language Technology Workshop. IEEE Xplore, 2011:19-24.

[3]. Zhang X, Wang H. A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding[C]. IJCAI, 2016.

[4]. Haffner P, Tur G, Wright J H. Optimizing SVMs for complex call classification[C]// IEEE International Conference on Acoustics. CiteSeer, 2003:I-632-I-635 vol.1.

[5]. Schapire R E, Singer Y. BoosTexter: a boosting-based system for text categorization[J]. Machine Learning(39):135-168.

[6]. Raymond C, Riccardi G. Generative and discriminative algorithms for spoken language understanding[C]//Interspeech. 2007: 1605-1608.

[7]. Mesnil G, Dauphin Y, Yao K, et al. Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding[J]. Audio Speech & Language Processing IEEE/ACM Transactions on, 2015, 23(3):530-539.

[8]. Yao K, Peng B, Zweig G, et al. Recurrent conditional random field for language understanding[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2014:4077-4081.

[9]. Liu B, Lane I. Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling[J]. arXiv preprint arXiv:1609.01454, 2016.

[10]. Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735.

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

[12]. Mesnil G, He X, Deng L, et al. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding[C]//Interspeech. 2013: 3771-3775.

[13]. Tur G, Hakkani-Tur D, Heck L, et al. Sentence simplification for spoken language understanding[J]. 2011, 125(3):5628-5631.

[14]. Guo D, Tur G, Yih W T, et al. Joint semantic utterance classification and slot filling with recursive neural networks[C]// Spoken Language Technology Workshop (SLT), 2014 IEEE. IEEE, 2015:554-559.

[15]. Zeiler M D. ADADELTA: an adaptive learning rate method[J]. arXiv preprint arXiv:1212.5701, 2012.

作者简介

孙鑫（1995--），男，博士生，研究领域为自然语言处理。
Email：sunxwith@163.com

王厚峰（1965--），男，教授，博士生导师，研究领域主要包括观点挖掘、问答系统。
Email：wanghf@pku.edu.cn