第 3 章　基于单模型的意图分类

3.1　引　　言

意图分类是许多自然语言理解任务和对话系统的关键预处理步骤，因为它允许将特定类型的话语分配到特定的子系统中。目前，在已有的相关研究工作中，已经探索了许多类型的神经网络(Neural Network，NN)架构，这些神经网络模型包括前馈神经网络和循环神经网络的结构。其主要目的是实现词汇的意图分类，并进一步发现未知意图。在本章中，将对两种领域的意图分类任务的神经网络模型进行更全面的介绍和比较，包括长短时记忆网络和门控循环单元网络。此外，鉴于之前的工作局限于相对较小和可控的对话数据集，因此，本章介绍基于从 Cortana 个人助理应用程序获得的大型对话数据集的测试实验比较情况。

本章首先将前馈神经网络、循环神经网络和门控网络(如 LSTM 和 GRU)相互比较。其次，本章比较了标准的单词向量模型和将单词编码为字符 n 字集的表示，以缓解超出词汇表的问题。实验结果表明，在几乎所有情况下，标准词向量优于基于字符的词表示。通过将神经网络模型的分数与 N-gram 语言模型的对数似然比进行线性组合，得到最佳分类结果。

使用深度神经网络模型的原因是由于之前基于 N-gram 的分类方法面临两个基本的问题：由于 N-gram 的时间范围有限及其稀疏性的特点，因此需要大量的训练数据来进行良好的泛化。选择建模的 N-gram 越长，稀疏性问题就越严重。为了解决数据的稀疏性，可以尝试为 N-gram 引入外部训练数据[61]，但这些方法最终受到了 N-gram 分布的领域特定特性的限制(从外部数据训练的模型通常不能通用化)。神经网络语言模型[62]能够使外部数据进行单词嵌入训练，并对模型进行改进。

然而，神经网络语言模型(Neural Network Language Models，NNLM)和标准的 N-gram 语言模型一样面临着一个问题：N-gram 有限的时间范围可能使其无法对依赖于时间上下文关系的文本内容进行有效的意图分类。基于循环神经网络(RNN)和长短时记忆(LSTM)[63]的模型在少量数据任务上确实优于标准的 N-gram 模型，但它们如何与神

经网络语言模型进行比较性研究仍是一个未解决的问题。由于之前的对话意图分类任务中标准单词向量的比较并不确定，因此在大量数据环境下研究基于字符 N-gram[64] 的单词表示的分类方法是必要的。

本章的目的是比较前馈神经网络、循环神经网络和门控网络模型（如 LSTM 和 GRU 模型）在小数据集和大数据集上的分类结果，以确定哪个模型在哪个场景下工作得最好。此外，将标准的单词嵌入特征表示与基于字符的 N-gram 特征表示进行实验对比，后者有望更好地泛化语料库中不可见的单词。

3.2 不同神经网络模型的对比

3.2.1 基线系统

本节使用两种基于 N-gram 特征的分类器架构作为对比性的基线实验：第一个基线实验是一对特定类型的 N-gram 语言模型，每个模型计算一个类似然，然后将这些可能性的对数比归一化为话语长度（词的数量），以获得阈值检测得分[61]；另一个基线实验对比方法是 Boostexter 算法[65-66]，该算法的输出分数也可以作为阈值分割的检测分数。

3.2.2 基于神经网络语言模型的话语分类器

图 3.1 是文献[62]中首次提出的 NNLM 基线模型系统的体系结构。它是一种基于神经网络的语言模型，此外，也是一种传统语言模型的替代，在文献[67-68]中首次被引入。与传统语言模型不同的是，它将单词编码为 n 维向量，根据任务的不同，标准维数从 100 到 1000 不等。在话语分类中，基于两个连续的单词预测对话意图来训练 NNLM。每句话的总分计算如下：

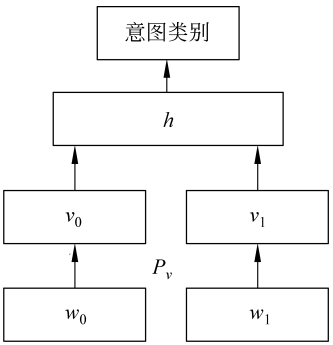

图 3.1 标准神经网络语言模型

$$P(L \mid \boldsymbol{w}) \approx P(L_1, L_2, \cdots, L_n \mid \boldsymbol{w}) = \prod_{i=1}^{n} P(L_i \mid \boldsymbol{w})$$

$$\approx \prod_{i=2}^{n} P(L_i \mid w_{i-2}, w_{i-1}, h_i)$$

$$= \prod_{i=1}^{n} P(L_i \mid h_i) \tag{3-1}$$

3.2.3　基于 RNN 的话语分类器

循环神经网络语言模型(RNNLM)[17]通过一系列隐藏单元对整个句子进行时序建模,其效果优于基于马尔可夫(有限记忆)假设的模型。与神经网络语言模型[67]类似,该模型将单词映射为一个密集的 n 维单词嵌入。隐藏状态 h_t 是以当前嵌入、前一个隐藏状态和偏置的函数。

$$h_t = \sigma(W_t h_{t-1} + v_t + b_h) \tag{3-2}$$

一般情况下,前馈语言模型中单词嵌入的最佳维数小于前馈语言模型的一半。

为适用于对话意图分类,使用对话意图类型标签训练一个 RNN 模型,如图 3.2 所示。RNN 根据 h_t 中存储的信息对对话意图进行分类。在测试时,属于一个对话意图标签的概率计算为

$$P(L \mid \boldsymbol{w}) \approx P(L_1, L_2, \cdots, L_n \mid \boldsymbol{w}) = \prod_{i=1} P(L_i \mid \boldsymbol{w})$$

$$\approx \prod_{i=1}^{n} P(L_i \mid w_i, h_{i-1}) = \prod_{i=1}^{n} P(L_i \mid h_i) \tag{3-3}$$

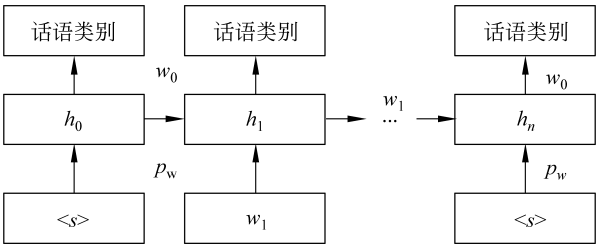

图 3.2　RNN 分类器模型

3.2.4　基于 LSTM 和 GRU 的话语分类器

在理想情况下,一个进行对话意图分类的模型应该为每个话语预测一个类别标签。在之前的实验中,RNN 模型在话语级预测单个标签的性能并不好,这可能是由于梯度消失的问题。对于这种情况尝试使用长短时记忆网络进行话语分类。

3.3　实　　验

3.3.1　数据集和评价指标

本实验的数据集采用文献[69-70]中使用的 ATIS 语料库。训练集由来自 ATIS-2

和 ATIS-3 的 A 类(上下文无关)部分的 4978 个对话,以及来自 ATIS-3 Nov93 和 Dec94 数据集的 893 个测试话语组成。语料库有 17 种不同的意图,将其映射到“飞行”与“其他”意图分类任务。输入有两种情况:一种只使用原始的文本单词;另一种称为自动标记(Auto Tagged),用短语标签(如 CITY 和 AIRLINE)替换实体,这些标签来自标记器[71]。

3.3.2 实验设置

神经网络语言模型分类器使用 500 维的单词嵌入和 1000 个隐藏单元的隐藏层,因为这样的实验室设置条件是在之前的工作和对两个数据集的实验中最优的。可能是由于单词嵌入大小的原因,早期的一些实验结果较差。RNN、LSTM 和 GRU 模型在两个语料库上都使用了一个 200 维的单词嵌入和单词哈希的数据结构。此外,LSTM 和 GRU 还包括一层 15 维的隐藏和存储单元。在不包含词嵌入的情况下,LSTM 模型比 RNN 多出约 150%的参数,而 GRU 比 RNN 使用的参数少。然而,对于大词汇量的任务,嵌入的大小超过了其他参数的数量,所以只使用最佳结果的模型结构。对于单词哈希,使用并行的三字母组合和二字母组合表示。例如,描述 cat 的集合是 #c、ca、at、t#、#ca、cat 和 at#。

一个训练好的系统通常使用截断的时间反向传播(BPTT)、正则化和梯度裁剪的组合。由于语料库的语言长度通常在 20 以下,发现使用梯度裁剪或截断的 BPTT 没有改善。此外,正则化及更高级的修改如 Nesterov 动量[72]的效果是微乎其微的,而简单的随机梯度下降在 Cortana 语料库上产生了较好的结果。

尽管 ATIS 数据集上的意图分类相对容易,但本项研究工作发现最终结果对初始参数很敏感,对于循环神经网络(RNN)和长短期记忆网络(LSTM)的权重服从正态分布 $\mathcal{N}(0.0,0.4)$分布,而 LSTM 门控权值除了门偏差是一个大的正值(大约 5)之外,都来自相同的分布。此外,借鉴前期工作[73]中一种有效的启发式方法,即在训练之前,计算 10 个随机种子在固定集合上的交叉熵,并从中选出交叉熵最小的那一个。

在 ATIS 数据集的实验中,初始学习率为 0.01,对于循环神经网络,动量参数为 3×10^{-4};对于 LSTM 模型,动量参数为 3×10^{-5}。验证集上的交叉熵在每个例子中减少到 0.01 以下时,学习率就会减半,并以相同的速度继续,直到达到相同的停止点。如前所述,使用 10 个不同初始化的最佳初始交叉熵。

在较大的 Cortana 数据集上,不需要动量参数。学习率参数与 ATIS 相似,除了验证集上的交叉熵改变小于 0.1%时学习率下降。由于更大的数据集包含更多的训练数据,实验结果表明,所有模型的随机初始化之间的方差都要小得多。

对于模型组合和评价,本实验使用线性逻辑回归(LLR)校准所有模型得分,并在适用

的情况下合并多个得分[74]。为了估计较小的 ATIS 数据集上的 LLR 参数，将数据集切分为 9 个大小相等的测试数据分区，依次训练除一个以外其余所有分区，并在所有分区循环。然后在整个测试集上合并得分，使用相同的错误率(EER)进行评估。

在 Cortana 数据集的实验中，开发集用于估计模型组合的 LLR 权重，然后在测试集上评估 LLR 权重，同样使用错误率进行评估。

3.3.3　实验结果

表 3.1 展示了不同神经网络结构下的 ATIS 意图分类结果。在标准和自动标记设置中，神经网络语言模型(NNLM)比所有其他模型都差，包括单词-三元组增强模型。LSTM 模型(LSTM-word、LSTM-hash)在常规设置中表现更好，而 GRU 模型(GRU-word、GRU-hash)在自动标记设置中表现更好。GRU 和 LSTM 模型的性能明显优于其他方法，从表 3.2 可以看出，两者的标准差都在一定范围内。

表 3.1　ATIS 意图分类结果

系　　统	EER/%	AutotagEER/%
Word 3-gram LM	9.37	6.05
Word 3-gram boosting	4.47	3.24
NNLM-word	6.05	4.03
RNN-word	5.26	2.45
RNN-hash	5.33	2.81
LSTM-word	2.45	1.94
LSTM-hash	2.88	2.81
GRU-word	3.24	1.58
GRU-hash	3.24	2.02

表 3.2　ATIS 意图分类详细结果

系　　统	EER/%	AutotagEER/%
平均误差		
NNLM-word	5.83±.238	4.15±.324
RNN-word	4.86±.919	3.50±.775
RNN-hash	4.32±.917	2.64±.324
LSTM-word	3.38±.986	2.22±1.01
LSTM-hash	4.05±1.13	2.64±1.02
GRU-word	4.44±1.69	3.58±1.24
GRU-hash	3.79±1.00	2.63±1.08

续表

系　　统	EER/%	AutotagEER/%
原始错误		
NNLM-word	6.05(5.61)	4.03(3.60)
RNN-word	3.95(3.95)	2.45(2.45)
RNN-hash	3.59(3.24)	2.45(2.09)
LSTM-word	2.81(2.45)	2.02(1.30)
LSTM-hash	3.24(2.88)	2.02(1.22)
GRU-word	3.24(1.70)	1.30(1.30)
GRU-hash	3.24(1.30)	2.02(2.02)

表 3.2 展示了所有模型的均值和标准差结果。到目前为止，前馈神经网络的标准差是所有结果中最低的，性能也是最差的。一般来说，门控网络比循环网络具有更高的方差。此外，在 10 个随机种子中使用最优种子交叉熵是一种合理的选择最佳随机种子的方法。

如表 3.3 所示，在更大的数据集中，结果更显著。对于 ATIS 数据集，模型从最坏到最优的相对排序为 NNLM、RNN、LSTM 和 GRU，其中 LSTM 和 GRU 的性能大致相同。此外，单词哈希和单词向量的表现差不多，两者在门控模型上表现更差，在循环模型上表现更好。保存字符 N-gram 所需的存储空间比单词向量少 1/3。

表 3.3　Cortana 域分类结果

系　　统	EER/%	ComboEER/%
Word 3-gram LM	7.37	—
Word 3-gram boosting	7.29	—
NNLM-word	9.33	7.30
RNN-word	7.99	6.80
RNN-hash	7.76	6.87
LSTM-word	6.86	6.56
LSTM-hash	7.11	6.63
GRU-word	6.78	6.46
GRU-hash	7.08	6.64

图 3.3 展示了两种 ATIS 测试条件和 Cortana 任务的检测误差权衡曲线。在大型数据集(Cortana)上，观察到非常笔直和平行的权衡曲线，这意味着不同的系统在某一种分类错误上没有特定的优点或缺点。这也意味着，系统在性能方面的相对顺序与所选择的操作点无关。

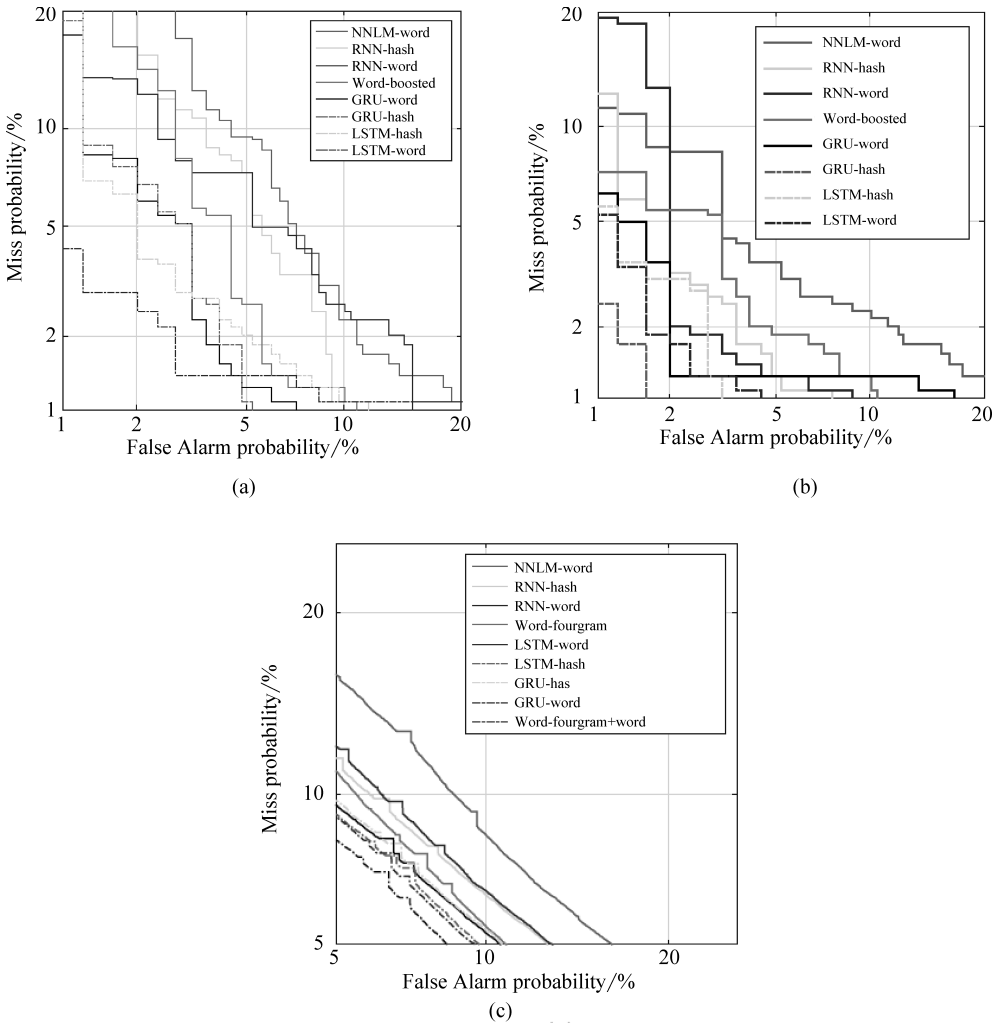

图 3.3　ATIS 数据集的 DET 曲线

3.4　本章小结

本章探究了各种神经网络架构在两种意图分类任务中的性能，使用了一个小规模语料库（ATIS）和一个大规模的现实生活数据集（Cortana）。根据分类效果性能对模型进行排序，依次如下：首先门控递归网络（GRU 和 LSTM）最好，性能大致相当；其次是普通循

环网络、前馈网络。门控单元网络的性能优于标准的 N-gram 基于语言模型的分类器或增强分类器。而 N-gram 语言模型通过逻辑回归组合可以进一步改进基于神经网络的分类模型系统。本章只考察词汇信息，以及从当前的对话语料中可以推断出什么。在今后的工作中，值得纳入非词汇信息（如韵律信息）[75]，以及在要进行意图分类的语境之前的对话语境[76]。建模方面，在单词序列[76]上加入一个卷积层是一个很有前途的模型架构。