自然语言处理与生物识别

Natural Language Processing for Biometric Identification

2020 年 2 月 17 日

Springer

目录

Chapter 1
语言模型与主题模型

本章简单介绍自然语言处理中的语言模型和主题模型的概念，并其常见的算法，以及在语音识别中的应用。

1.1 语言模型

1.1.1 基本定义

语言模型（Language Model）用于计算语言序列 $w_1, w_2, \cdots, w_n$ 的概率，数学表示为 $P(w_1, w_2, ..., w_n)$，它是对语句的概率分布的建模。其最直接的应用就是判断一句话来自于人生成的语句的概率，例如在我们自然语言中，句子 “我去吃饭” 相比于 “吃饭去我” 的出现的概率更高，因此 $P(\text{“我去吃饭”}) > P(\text{“吃饭去我”})$。讲到这里，最直接的一个问题就是，如何计算 $P(w_1, w_2, ..., w_n)$ 呢? 我们下面介绍一种最基本的语言模型: n-gram 语言模型。

1.1.2 n-gram 语言模型

n-gram 语言模型是一种最基础的语言模型。根据链式法则 (Chain Rule)，公式 $P(w_1, w_2, ..., w_n)$ 可以得到:

$$P(w_1, w_2, \cdots, w_n) = P(w_1)P(w_2|w_1)\cdots P(w_n|w_1, \cdots, w_{n-1})$$

其中的每一项 $P(w_i|w_1,\cdots,w_{i-1})$，可以用以下公式来估计，即:

$$P(w_i|w_1,\cdots,w_{i-1}) = \frac{C(w_1,\cdots,w_{i-1},w_i)}{C(w_1,\cdots,w_{i-1})}$$

其中，$C(\cdot)$ 表示该序列在训练语料中出现的次数。但是，当序列长度很长时候，计算 $P(w_i|w_1,\cdots,w_{i-1})$ 比较困难，因此大家引入了马尔可夫假设 (Markov Assumption)，即假设当前词出现的概率只依赖于前 $n-1$ 个词，也就是:

$$P(w_i|w_1,\cdots,w_{i-1}) = P(w_i|w_{i-n+1},\cdots,w_{i-1})$$

根据 n 取值的不同，我们可以得到以下模型:

- Unigram: $P(w_1,\cdots,w_{i-n}) = \prod_{i=1}^{n} P(w_i)$
- Bigram: $P(w_1,\cdots,w_{i-n}) = \prod_{i=1}^{n} P(w_i|w_{i-1})$
- Trigram: $P(w_1,\cdots,w_{i-n}) = \prod_{i=1}^{n} P(w_i|w_{i-1},w_{i-2})$

1.1.3 n-gram 语言模型中的平滑技术

在计算 n-gram 时候，一个很重要的问题就是测试集中出现了训练集中未出现过的词而导致语言模型计算出的概率为零，我们称这些词为未登录词 (OOV)。平滑 (Smoothing) 技术就是为了缓解这类问题，常见的平滑技术有：拉普拉斯平滑 (Laplace Smoothing),、古德图灵法（good-turing)、线性减值法（Linear Discounting）等，感兴趣的读者可以深入阅读相关论文。

1.1.4 语言模型在语音识别中的应用

自动语音识别（Automatic Speech Recognition，ASR）是一种将人的语音转换为文本的技术，它是目前很多互联网产品如语音助手，语音搜索引擎等中必不可少的一部分。图 1-1给出了常见的语音识别系统的基本工作流程。其中基本可以分为以下几个模块:

- 数据预处理: 典型的预处理包含静音处理（Voice Activity Detection，VAD）等，用于去除其中的静音片段。

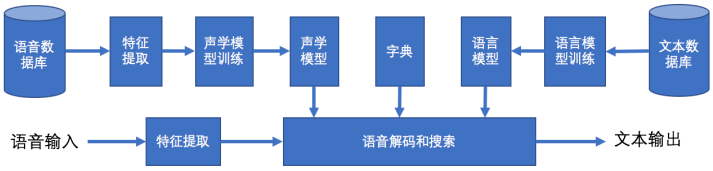


图 1-1 语音音识别基本流程

- 特征提取：将声音转换成包含声音信息的多维向量，常见的有 MFCC 等。
- 声学模型：主要是通过语音数据训练得到，其输出是音素等信息。
- 词典：字/词和音素之间的对应关系。
- 语言模型：也就是上文提到的语言模型部分，主要用于评估字或者词序列的概率。

语音系统首先将语音信号做特征提取工作，转化成诸如 MFCC 等特征来表示，然后使用语言模型和声学模型来解码，解码过程会产生很多候选 (Candidates)，最终最优的候选会被输出成为最终的结果。语言模型是其中很重要的一部分，它用于从根据语言统计规律评估声学模型给出的句子序列候选的概率，决定了最终输出的结果。

1.2 主题模型

1.2.1 基本概念

主题模型 (Topic Models) 是近些年来非常重要的一项技术，它被广泛应用于工业和学术界。在主题模型中，我们一般用 d 来表示要分析的文档，例如一篇文章或者一个网页等，而一个文档 d 通常由一系列词 $(w_1, w_2, ..., w_n)$ 组成，其中 w_n 是文档中的第 n 个词。多份文档共同构成了我们要分析的语料集，我们用 $\mathscr{D}$ 来表示，$\mathscr{D} = (d_1, d_2, ..., d_m)$ 组成，其中 d_m 是语料库中的第 m 个文档。主题一般用 z 来表示，它由一些词组成，同时也有该词在这个主题下的概率。主题模型泛指由一类可以从语料库

中抽取主题并利用这些主题表示文档的模型，常见的主题模型有 PLSA，LDA，以及各种 LDA 的变种，例如 SentenceLDA 等。在熟悉了这些基本概念之后，我们通过一种常见的主题模型 Latent Diriclet Allocation (LDA) 来认识主题模型。

1.2.2 常见的主题模型：LDA

2003 年 Blei 等人在《Latent Dirichlet Allocation》[66] 一文中提出了 LDA 模型。如图1-2所示，其中空心节点表示隐藏变量，实心变量表示客观测变量，整个模型具有 K 个主题，M 个文档和 N 个词。LDA 将文档的主题分布 $P(z|d)$ 看做随机变量 θ，并且假设 θ 从一个狄利克雷先验中产生。同时，由于训练数据之外的文档对应的主题分布 θ 可以从上述狄利克雷分布中产生，训练数据之外的文档的 θ 可以更自然地进行计算。

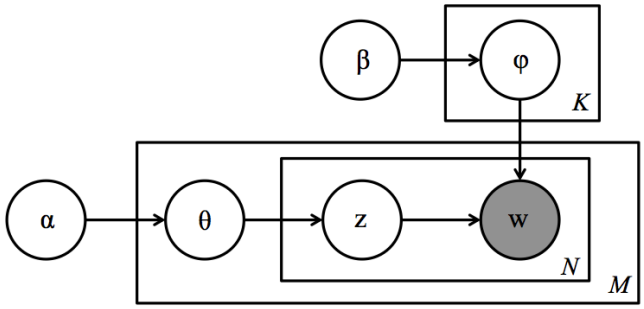


图 1-2 LDA 图模型

1.2.3 主题模型在语音识别中的应用：语言模型适配

语音识别系统中一个常见的问题就是，我们训练语言模型的语料和它实际线上应用的语料之间存在不一致，这种情况下，除了重新训练模型，有一种代价更小的方法就是语言模型适配（Language Model Adaptation）。

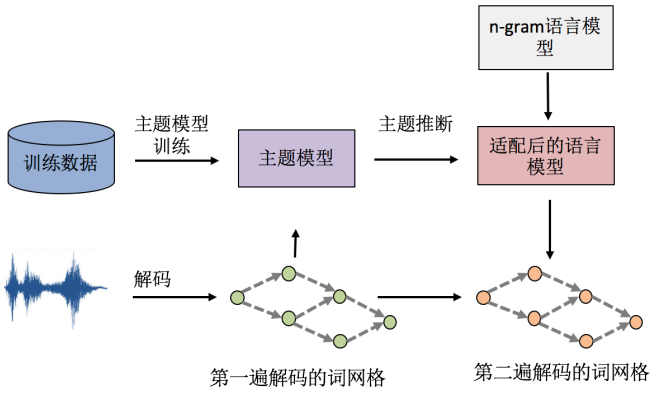


图 1-3 语言模型适配

语言模型适配指的是用实际应用的语料相关的信息，对语言模型做适配。图 1-3给出了其常见的工作流程，采用预先训练好的主题模型，我们对语音识别系统第一遍识别出来的词网格 (Word Lattice) 做主题推断，可以发现其语义级别的内容，同时它也可以作为一个 unigram 的语言模型 $P_{LDA}(w|\theta_d) = \sum_{k \in K} \varphi_{kw} \theta_{dk}$，对之前的 n-gram 语言模型就行适配：

$$P_d(w|C) = \lambda P_{LDA}(w|\theta_d) + (1-\lambda) P_{n-gram\ LM}(w|C) \tag{1.1}$$

其中 C 代表当前词 w 的上下文，λ 是一个权重参数，$P_{n-gram\ LM}(w|C)$ 是基础的 ngram 语言模型给出来的评估分数。这个新适配过的语言模型，可以用于语音识别系统，重新解码出新的词网格以及最终的结果。

Chapter 2
词法，语法与语义分析

2.1 词法分析

2.1.1 问题定义

词法分析是自然语言处理的第一步，要做 NLP 深层次分析，比如句法分析、语义分析，甚至 NLP 复杂应用的先决条件，就是首先进行词法分析。词法分析的核心是，将自然语言解析为一个个词的序列，并判断每个词的词性、专名信息，为后续分析做好准备。总的来说，在中文这种孤立语中，词法分析主要由分词、词性标注、命名实体识别 3 个子任务组成。在英语、阿拉伯语等屈折语中，词法分析一般还包括词根还原（word stemming）任务。忽略词根还原，词法分析可以归纳为 3 个具体的子任务：

- 自动分词（word segmentation）：是将连续的自然语言文本，切分成具有语义合理性和完整性的词汇序列的过程。
- 词性标注（Part-of-Speech tagging）：是指为自然语言文本中的每个词汇赋予一个词性的过程。
- 命名实体识别（Named Entity Recognition，也称专名识别）：是指识别自然语言文本中具有特定意义的实体，主要包括人名、地名、机构名、时间日期等。

如图2-1所示，将输入句子切分成一个个词汇，然后给每个词汇标记出名词、动词、介词等词性；并且识别出“2003 年 10 月 15 日”是一个时间专名，识别“杨利伟”是一个人名专名等。

图 2-1 词法分析示例

因为词法分析的自动分词、词性标注、专名识别本质上是类似的，所以它们的发展历史方也是类似的。总结起来，都大致经历词典匹配、机器学习、深度学习这 3 个发展阶段。

2.1.2 实现方案

2.1.2.1 词典匹配

基于词典匹配的实现步骤：

1. 词典构建：根据具体子任务的语言学知识构建词典。如果是自动分词，则收录常见的词条短语；如果是词性标注、专名识别，则收录当前常见词条对应的名词、动词、介词、专名类型等属性，并且保留相应词性、专名属性的概率。词典构建，通常需要人工收集、整理、离线更新，维护成本较高。
2. 词典匹配：扫描输入的所有子序列，如果当前子序列能够匹配词典中某个词条，则当前子序列属一个可能的候选。如果是分词，则当前子序列就可能为一个分词的词汇；如果是词性标注、专名识别，则当前子序列以相应的概率取词典中的词条属性。
3. 歧义消解：由于候选子序列之间存在歧义，所以基于词典匹配之上，需要加入一些启发式规则以解决歧义问题。常用规则主要包含最大前向匹配（Forward Maximum Matching，FMM）、最大后向匹配（Backward Maximum Matching，BMM）、最少切分、双向最大匹配、长片段优先等策略 [1][2]。

中文分词如图2-2所示。输入序列进行词典匹配之后，得到对应的 DAG 图，图中每条边都是词典词条，边的权重都为 1，图中每一种首尾

贯通的连接都是一种歧义切分的候选结果。比如由此只要求解 DAG 图的最优路径，则可以得到输入序列对应的分词结果。

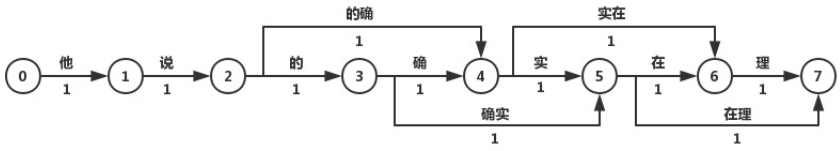


图 2-2 中文分词示例

很长一段时间内研究者都在对基于词典匹配方法进行优化，比如最大长度设定、字符串存储和查找方式以及对于词表的组织结构，比如采用 TRIE 索引树 [3]、哈希索引 [4] 等、AC 自动机 [5] 等结构方便快速查找。

基于词典匹配的优缺点：

- 优点：实现简单、运行速度快
- 缺点：面临词典收录更新困难、未登录词难处理；同时由于消除歧义策略过于简单，通常效果不甚理想。

2.1.2.2 机器学习

基于机器学习的实现步骤：

1. 转换为序列标注任务：词法分析 3 个子任务，都通过定义标注空间标签集，将具体任务转换为标准序列标注任务。以中文分词为例：转换方法为，标注每个字在其所属词中的位置。因为对于任何一个字来说，它可以是一个词的开始（Begin），一个词的中间（Inside），一个词的结尾（End），或者本身就是一个单字的词（Singleton），这也就是在分词序列标注中常用的 BIES 的分类。只需将输入序列的每个字标上 BIES 标签中的一个，就可以转换得到对应的分词结果。这种标注空间（模型状态空间）的划分在词性标注和专名识别任务上也很常用，也会有一些类似的变种，比如专名识别中常用 BIO 标签集。

2. 求解序列标注任务：传统序列标注模型主要包括隐马尔科夫模型（Hidden Markov Model）[6][7][8][9]、最大熵马尔科夫模型（Maximum Entropy Markov Model）[10][11][12]、条件随机场（Conditional Random Field）[13][14][15]、结构化感知机（Structural Perception Machine）[16][17] 等浅层模型。这些浅层模型的区别主要在于如何对待输入字序列和标签序列之间的概率，训练目标是最大联合概率似然、最大条件概率似然，还是最小化风险等。

总结来说：传统序列标注模型中，CRF 是集大成者。相比于 HMM，CRF 去除了输出独立性要求，对于整个序列内部的信息和外部观测信息都可以有效利用，可以更加有效建模上下文。相比于 MEMM，CRF 通过全局归一化（global normalization），避免了 MEMM locally normalized 导致的 label bias 缺陷。

以自动分词任务为例，则其序列标注任务定义为：定义标签集为 $L = \{B, I, E, S\}$，给定输入文本序列 $X = \{x_1, x_2, ..., x_n\}$，目标是求解最优标注序列 $Y^* = \{y_1^*, y_2^*, ..., y_n^*\}$：

$$y^* = \arg\max_{Y \in L^n} p(\boldsymbol{Y}|\boldsymbol{X})$$

预测时，使用维特比算法求解最优标注序列 Y^*；训练时，使用最大化条件似然来训练模型，其中，f_i 为人工定义的特征函数：

$$P_w(Y|X) = \frac{1}{Z_w(x)} \exp(\sum_i w_i f_i(y_i, y_{i-1}, x))$$

$$Z_w(x) = \sum_{y \in Y} \exp(\sum_i w_i f_i(y_i, y_{i-1}, x))$$

基于机器学习的优缺点：

- 优点：通过人工设计的特征工程，充分地挖掘了序列的上下文信息。模型的歧义消解胜过词典匹配的方法；同时具有很强的泛化能力，能够很好地处理未登录词问题。
- 缺点：特征工程不经需要耗费大量人力，而且需要大量语言学知识，设计和寻找有效特征存在较高门槛；另外，这些浅层模型，通常使用离散的 binary 特征，无法表达复杂先验，比如没法利用词向量。

2.1.2.3 深度学习

随着大数据、神经网络、深度学习的快速发展，很多研究提出利用前馈神经网络来解决词法分析 [18][19] 任务。总的来说，与前面基于机器学习的方法类似，也是把词法分析任务作为序列标注问题进行求解，只是把人工设计特征函数，改成了使用多层前馈神经网络进行自动特征抽取。

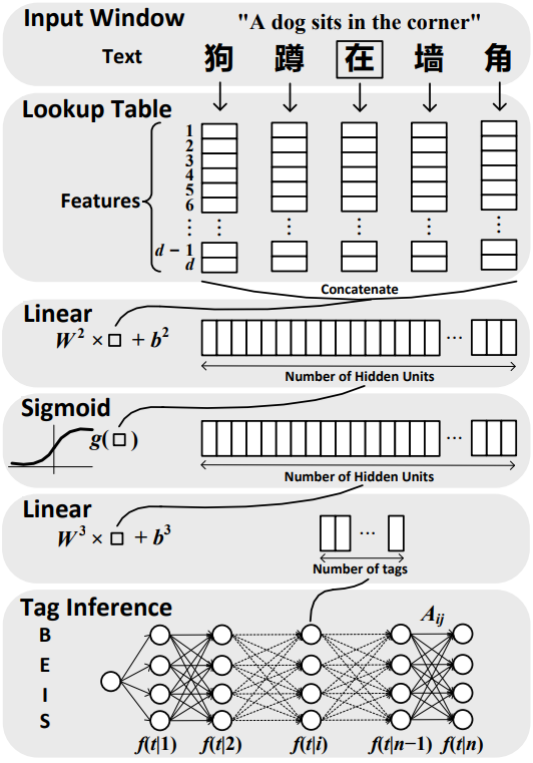


图 2-3 深度学习分词示例

如图2-3所示，网络的第一层输入句子中每个字的字向量，第二层将一个固定长度的字向量进行拼接，然后输入到标准前馈神经网络结构中，神经网络输出在标注集合上的 lattice，最后利用维特比算法进行解码，就可以得到最优标注序列。

后续有很多研究，深度学习框架基础上，对前馈神经网络进行改进，进一步探索了卷积神经网络 [20]、循环神经网络 [21]、递归神经网络 [22] 等复杂结构对词法分析任务的影响。

另外，传统的词法分析通常会把分词、词性标注、命名实体识别当成 pipeline 形式的进行处理，这样带来的一个问题就是错误传播。比如，分词的错误将会导致后续词性标注、专名识别、句法分析、语义分析出现错误。所以在学术界也有很多联合建模 [17][23][24] 方面的工作。联合建模的一大好处是词法分析与其他任务可以共享有用的信息，词法分析的时候也会考虑到其他任务的要求，其他任务也会考虑各种词法分析的可能性，通常可以再全局上取得最优解。但是随之而来问题是搜索的复杂度往往会显著提高：需要更有效的搜索剪枝机制在控制复杂度的同时，不对搜索的结果产生显著影响。

值得一提的是，2018 年 Peters 等证明在预训练语言模型 [25][26][27]，在各种 NLP 任务中提供比最先进的技术更大的改进。可以将语言模型嵌入作为特征，使用目标任务数对语言模型对进行微调 [28][29]，通常就能达到或超过传统结果。由于语言模型只需要无标记的数据便可以进行学习，因此对于标记数据稀缺的低资源场景，预训练语言模型尤其有用。

总结起来，基于深度学习方法的优点是通过深度神经网络自动学习多层特征抽象，避免了复杂的特征工程；模型的歧义消解、泛化能力通常都很好。只是，深度学习模型需要训练充分，通常对数据量、计算量都有较高的要求。

2.1.3 应用案例

通常 NLP 中深层次的语法语义分析通常都是以词作为基本单位，所以词法分析都是这些深层次分析的基础和先决条件。词法分析作为自然语言处理的第一步，是下游所有分析任务的基础。这些下游应用小句法分析、语义分析，大到对话系统、文本分类、自动摘要、机器翻译 [30]、信息检索、搜索引擎、语音合成等等。几乎只要有利用到自然语言处理技术的地方，词法分析都是不可或缺的基础技术。

比如在搜索引擎中，用户输入一个表述需求信息的查询字段，系统回复一个包含所需要信息的文档列表。其核心技术在于索引构建和相关性计算。

索引构建：首先需要对互联网海量文档进行，分词、词性标注、专名识别；利用词法分析的结果，以词为粒度，为所有文档建立倒排索引。

相关性计算：首先需要对用于查询字段进行词法分析，找出表达用户需求的核心词汇；然后以这些核心词汇为 key，去拉取相应的倒排索引，获取初步筛选后的文档；接着，为了要提高返回结果的相关性，需要计算用户查询和初筛文档之间的相似度，取相似度较好的文档返回。其中计算用户查询和文档之间的相似度，通常利用 BM25、语义计算等技术，而它们也都是以词为单位进行的，由此看出词法分析在搜索引擎中的重要性、基础性。

在比如在文本分类和情感分类中，在预定义分类体系下，输入给定文本，抽取文本特征，将给定文本于一个或多个类别相关联的过程。其中，抽取文本特征的时候，通常也都是先进行词法分析，先获得给定文本的词汇、词性、专名等信息，然后以词汇为单位，以词汇的词向量、词性专名等信息为特征，利用机器学习、深度学习的分类模型进行预测。

2.2 语法分析

2.2.1 问题定义

短语结构语法（Constituency Structure Grammar）和依存关系语法（Dependency Grammar）是现在常见的两种语法关系。短语结构语法又叫上下文无关文法 (Context-Free Grammars，CFGs)，它从一个特殊的初始符号出发，不断的应用一些产生式规则，从而生成出一个字串的集合（如句子）。产生式规则指定了某些符号组合如何被另外一些符号组合替换。它呈现一个树分类关系，句法根据一定的规则进行转换分析。每一个词的转换都是需要按照设定的树值规则进行目的性的转换。

依存语法 (从属关系语法) 是由法国的语言家 Lucien Tesnière 提出的 [31]，它将句子各个词语之前的搭配关系描述成预先定义好的依存关系。它基于一个基本假设：句法结构本质上包含词和词之间的依存（修

饰）关系。一个依存关系连接两个词，分别是核心词（head）和依存词（dependent）。依存关系可以细分为不同的类型，表示两个词之间的具体句法关系。比如主语依赖于谓语（SBV），宾语也依赖于谓语（VOB）以及定语依赖于名词性短语（ATT）等。依存句法认为“谓语”中的动词是一个句子的中心，其他成分与动词直接或间接地产生联系。

除了以上介绍的两种句法体系外，国内外都开展了对句法分析的研究。不论是国外的链语法 (Link Grammar)、组合范畴语法（Combinatory categorial grammar，CCG）等，还是国内黄曾阳教授提出的 HNC 理论 (Hierarchical Network of Concept)[32] 都是目前行业内常用的语法，只是由于设定区域的不同，所以使用有一定的局限性。

短语结构分析的语法集是由固定的语法集组成，较为固定和呆板，依存语法则更加的自由。另外依存语法树标注简单且 parser 准确率更高，再加之通用依存数据集（Universal Dependencies Treebanks）的发展，依存语法分析受到专家学者普遍的青睐，得到越来越多的应用。这里也将着重介绍依存语法分析。

图 2-4 LTP 依存句法分析样例

例如，从上述例子中我们可以看到，句子的核心谓词为“强调”，主语（SBV）是“他”，强调的宾语（VOB）是“紧迫性”，“紧迫性”的修饰语（ATT）是“武汉疫情控制”。

2.2.2 实现方案

依存句法分析方法主要可以分为两种，一种是基于图的方法（Graph Based），一种是基于转移（Transition Based）的方法。基于图的方法先

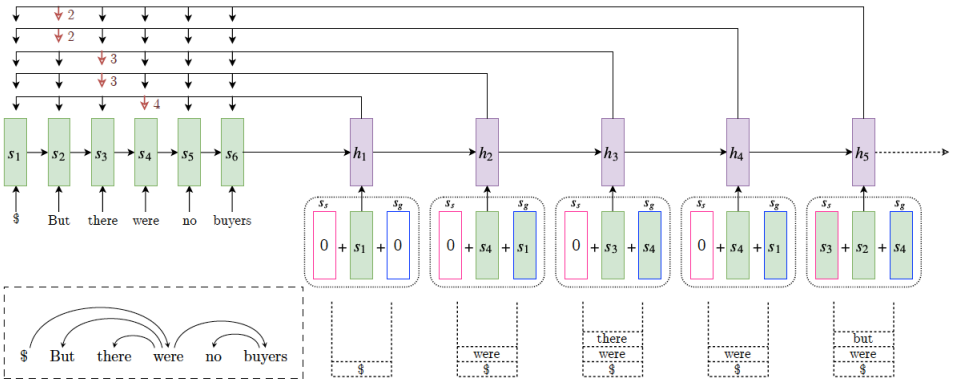
建立句子中所有词语的全连接图，然后求图中的最大生成树。两种方法中更主流的算法是基于转换的依存句法分析，基于转移的方法将依存树的构成过程建模为一个动作序列，将依存分析问题转化为寻找最优动作序列的问题。通过 SHIFT，LEFT_ARC，RIGHT_ARC 三个动作来将序列转换为树结构。一次分析任务 $C=(s,b,A)$ 由一个 Stack 栈 s，一个 buffer 缓冲区 b，一系列依存弧列表 A 构成。初始化栈 s 里面只包含一个 *root* 元素即根元素，s_1 代表栈顶元素，s_2 表示栈顶第二个元素。缓冲区 b 是一个队列，里面包含了要解析的一句话的序列，A 为空。一条依存弧有两个信息：动作类型 + 依存关系名称 l。l 视依存句法语料库中使用了哪些依存关系 label 而定。动作 SHIFT 将缓冲区 b 中最上面的一个词 b_1 移到 stack 中，即不建立依存关系，只转移句法分析的焦点，即新的左焦点词是原来的右焦点词，依此类推。LEFT_ARC，即添加栈顶两个词 s_1，s_2 之间的依存边，方向为 $s_1 \rightarrow s_2$，并且将 s_2 从栈 s 中删除。RIGHT_ARC，即添加栈顶两个词 s_1，s_2 之间的依存边，方向为 $s_2 \rightarrow s_1$，并且将 s_1 从栈 s 中删除。

比如图2-4中，首先模型判断执行 SHIFT 动作将“他”移到栈 s 上，然后再执行 SHIFT 动作将“强调”移到栈 s 上，接着模型判断这两个词之间有依赖关系，且方向为为“强调 $\rightarrow$ 他”，执行 RIGHT_ARC 动作在 A 中添加一条依存弧，且将“他”从栈 s 中删除。接着模型判定执行 SHIFT 动作将“了”移到盏 s 上，然后模型判断执行 LEFT_ARC 操作，即添加“强调”和“了”之间的依存弧“强调 $\rightarrow$ 了”到 A 中且将“了”移出栈 s。如此往复，直到最后当缓冲区 b 为空和栈 s 只有 *root* 时结束。训练模型的主要目标即寻找一个分类器，当给定一个 Configuration (当前的 stack，buffer，依存弧列表）时预测下一步转移的操作类别。

基于转移的解析过程是线性的，动作步骤随句子长度线性增长，而基于图的方法需要在全图上做搜索，所以时间复杂度上基于转移的方法会有优势。但是基于转移的方法在解析的每一步都只是利用局部信息，会导致错误传播，性能比基于图的略差。

近几年，分别出现了针对这两种不同方法的神经网络模型。比如基于图的 [33, 34, 35]，直接用神经网络来预测每两个词之间存在依存关系的概率，得到一个全连接图，图上每个边代表了节点 a 指向节点 b 的概率，然后使用 MST 等方法来来将图转换为一棵树。概率的计算可以简单的使用节点 a 和节点 b 的 embedding 向量做向量运算，也可以使用复杂的多

层 GNN 网络迭代更新。基于转移的如 [36, 37, 38]，通过两个 LSTM 来分别建模 stack 状态、buffer 状态，使用第三个 LSTM 网络或者 Pointer 网络来建模动作序列。



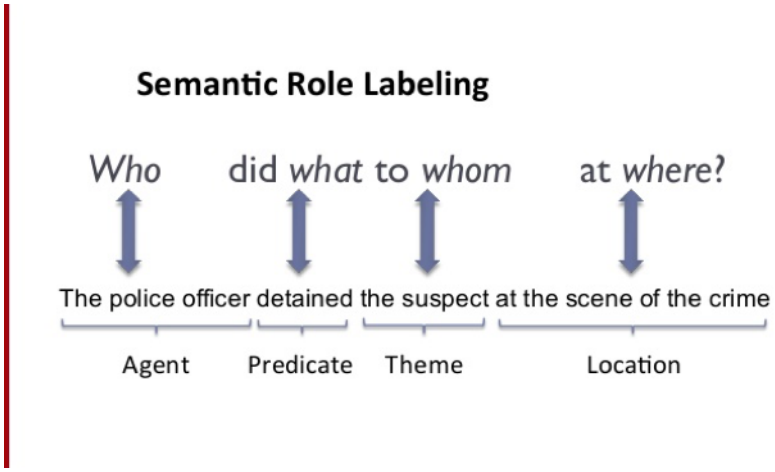
2.2.3 应用案例

依存语法分析在信息检索，评价抽取和情感分析等 NLP 任务上都有很多应用。比如“谢霆锋的儿子是谁”和“谢霆锋是谁的儿子”两句话如果不使用依存语法分析，很有可能就返回了一样的结果。依存句法分析能够更直接地通过语法结构的规则约束筛选出可能正确结果，提升相应任务的准确性。又比如“深圳大学非常漂亮，学生都很聪明”，这里“漂亮”形容的是“深圳大学”，“聪明”形容的是“学生”，通过依存句法分析，就可以抽取出对应的搭配。再比如“我家音响声音很大”和“我家洗衣机声音很大”，两者在情感上前者是正评价，后者是负评价，需要使用依存句法分析来识别“声音很大”的修饰对象。

常用的中文依存句法分析的工具有复旦大学 fnlp[1]，斯坦福大学 Stanford CoreNLP[2]，Hanlp[3]和哈工大 LTP[4]。

2.3 语义分析

在自然语言处理领域，语义分析涉及在某种程度上理解单词、短语、句子或文档的意义。传统狭义语义分析主要包括语义消歧（word sense disambiguation）[39] 和语义角色标注（semantic role labeling）[40, 41]。语义消歧指在给定文本上下文中确定多义词语的含义，例如，“他买了一台新苹果，用来修图更方便了”中，苹果一次指代苹果电脑，而非水果。语义角色标注是给词语和短语标注其在上下文文本中的含义的过程，典型标注标签包括主体、意图、结果等。语义角色标注是面向任务型对话系统中核心组件自然语言解析模块的基础技术之一。随着深度神经网络在自



然语言处理领域的应用和研究，基于神经网络的语义分析得到了越来越多的应用和发展，并成为了驱动神经机器翻译 [42, 43]、阅读理解 [44]、对话系统 [45, 46] 的最基础和核心的技术。基于神经网络的语义分析广义上

[1] https://github.com/FudanNLP/fnlp

[2] https://stanfordnlp.github.io/CoreNLP/

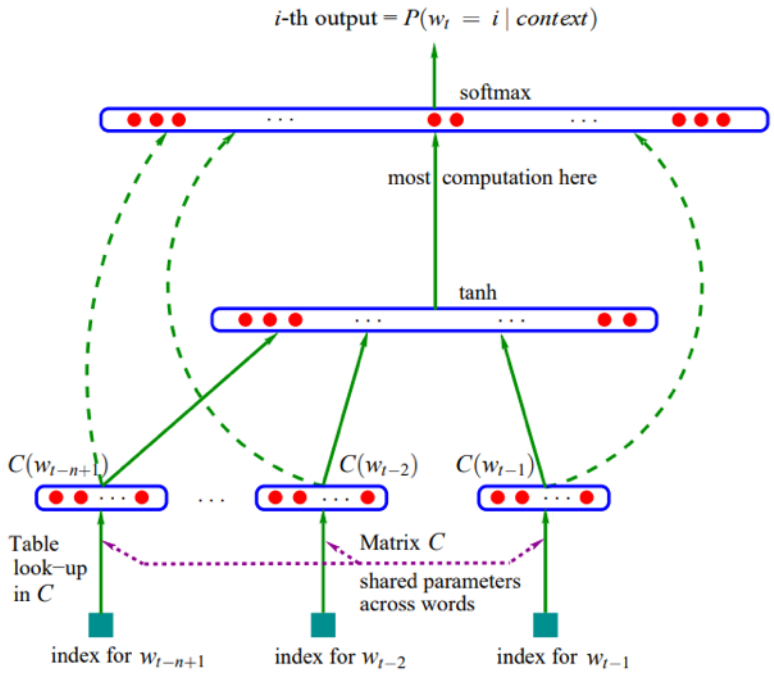
[3] https://github.com/hankcs/HanLP

[4] https://github.com/HIT-SCIR/ltp

可分为语义表示和语义匹配。其中语义表示任务将词语和短语镶嵌到高维向量空间中，称作词向量，作为 CNN[47]、RNN[48] 和 Transformer[49] 等模型的底层输入，根据任务需要，可以灵活的实现文本分类 [50, 51]、机器翻译、阅读理解、对话系统、文本摘要 [52, 53] 等自然语言处理任务，逐步替代或部分替代了传统以 ngram one-hot 向量作为底层表示的方法。语义匹配任务更多关注句子和篇章层级的语义的相似性，例如，寻找给定语料库中与查询语句语义层面最相似的句子。语义匹配任务在信息抽取、对话系统、问答系统都有广泛的应用。

2.3.1 语义表示

语义表示中词向量的概念可以追溯到 Bengio 的著作 [54]。作者在文中提出了一种基于神经网络的语言模型，创新的提出将每一个词表示成一个高维的向量，而后使用神经网络计算给定上文词语后，下一个词出现的概率分布，最大化训练语聊出现的概率。训练语料是单词序列 $w_1,\cdots,w_t, w_t \in V$，其中 V 是全体单词的集合。语言模型的目标是找到一个模型 $f(w_t,\cdots,w_t-n+1) = P(w_t|w_1^{t-1})$，即在给定前 $t-1$ 个词的情况下，对第 t 个词的概率分布建模。Bengio 提出把词表示成实数向量，$C(i) \in R^m$，即每一个词对应一个维度为 m 的实数向量，映射 C 可以用一个 $|V| \times m$ 的实数矩阵表示。在第二步中，作者提出使用神经网络 g 来建模给定前缀序列下一个词的概率。其中，将词表示成高维实数向量的方法是词向量以及一系列神经网络在自然语言处理应用的开端。通常，我们会设计一个仅与文本结构或上下文相关的（无监督）任务作为目标，拟合训练数据，得到词向量表示。在 Bengio 之后，Word2Vec[55], Glove[56], ELMo[57] 等一系列词向量方法出现并逐渐成为自然语言处理的标准工具。语义表示不仅限于词的层面，广义的语义表示在现代基于神经网络的自然语言处理中应用广泛。例如，神经机器翻译通常采用编码器译码器（Encoder-Decoder）[58] 结构，其中编码器的输出可以看做是翻译模型对输入文本的语义表示，译码器基于该表示，生成目标语言的文本。再如，闲聊型对话系统中，需要针对聊天的上文给出合适的回答。闲聊对话系统可以分为生成式对话系统（Generation-based）[59] 和选择式对话系统（Selection-based）[46]。选择式对话系统依赖语义匹配方法。典型的

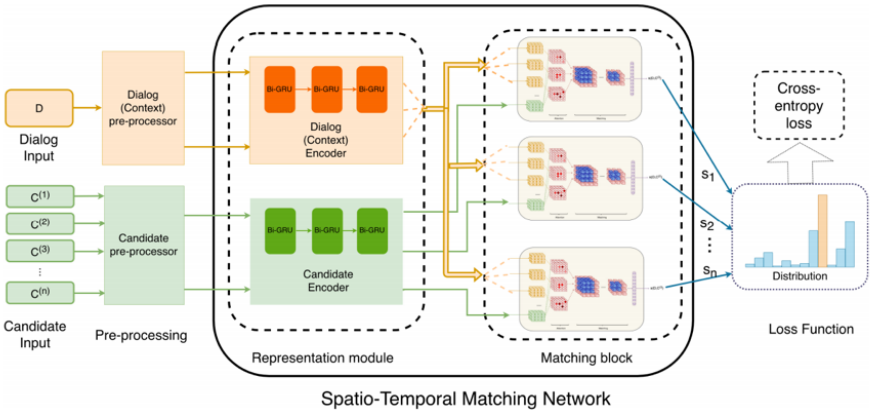


生成式闲聊对话系统也采用编码器译码器结构，其中编码器的输出是对聊天上文的总结性表示，可以认为是广义上的聊天上文的语义表示，用于在译码器中生成合适的回答语句。此外，随着自然语言处理预训练模型的发展，Bert[60]，GPT-2[61]，MASS[62] 等预训练模型（Pretrained models）在阅读理解等任务上大放异彩。这些模型可以看做是多个联合任务共享底层语义表示的学习，通常使用与文本结构相关的目标作为训练目标，设计神经网络结构表示文本。在应用时，这些预训练获得的表示仅需在少量标注数据上进行优化，即可获得优秀的子任务模型。通常，编码器和预训练模型采用 CNN、RNN、Transformer 等结构处理文本序列，在 CNN 和 RNN 中通常还会使用注意力（Attention）机制，得到文本片段的语义表示。

2.3.2 语义匹配

语义匹配任务在语义表示的基础上，对文本片段之间的相似度或相关度给出量化指标，语义越相似的片段匹配分数越高。语义匹配可大致分为基于表示的匹配（representation-based matching）[63] 和基于交互的匹配（interaction-based matching）[64, 65]。基于表示的匹配方法注重对表示层的构建，将待匹配的对象通过语义表示的方法转换成等长的向量，并在此基础上进行匹配度计算。常见的匹配度计算方式包括 cosine 函数，和神经网络匹配。Cosine 函数直接计算待匹配对的语义表示向量间的 cosine 值，通常 cosine 值越大代表待匹配对的相似度或相关度越高。这种方法不需要额外的训练数据，实现简单并且高效，在工业中广泛应用。神经网络匹配方式使用一个额外的神经网络结构，将待匹配对的语义表示向量作为输入，计算语义匹配分数。这种方法更加灵活，可根据数据定制匹配结构。但需要额外的标注数据进行训练，才能得到可用的匹配模型。基于交互的匹配方法通常会保留待匹配对的序列信息，不会在表示

层将文本转换成唯一的一个整体向量表示，而保留一个向量序列，用于接下来的交互匹配过程。例如将句子表示成一个与句子等长的向量序列。该向量序列可以使用语义表示的方法得到，例如使用 RNN 对文本序列进行建模。得到待匹配的序列对后，可以对序列中每个位置的向量计算与待匹配序列中向量的相关度。对每一个位置使用相同的方式计算相关度，可以得到一个匹配矩阵（matching matrix）。匹配矩阵包含了更细致的局部文本交互信息，在交互矩阵的寄出上，我们可以构建神经网络结构计算最终匹配得分，去拟合目标得分。下图 [65] 展示了一个典型的基于交互的语义匹配方法在选择式对话系统中的结构。其中 Representation module 对应语义表示部分，matching block 对应交互匹配部分。


D
Dialog Input
Dialog (Context) pre-processor
Bi-GRU
Bi-GRU
Bi-GRU
Dialog (Context) Encoder
C(1)
C(2)
C(3)
C(n)
Candidate Input
Candidate pre-processor
Pre-processing
Candidate Encoder
Representation module
Matching block
Cross-entropy loss
s1
s2
sn
Distribution
Loss Function
Spatio-Temporal Matching Network

References

[1] WONG P-K, CHAN C. Chinese word segmentation based on maximum matching and word binding force[C] // Proceedings of the 16th conference on Computational linguistics-Volume 1. 1996 : 200 – 203.

[2] WEICHUN H, JIANJIAN J. Research on Longest Backward Segmentation for Context[C] // 1st International Workshop on Cloud Computing and Information Security. 2013.

[3] YANG L, XU L, SHI Z. An enhanced dynamic hash TRIE algorithm for lexicon search[J]. Enterprise Information Systems, 2012, 6(4) : 419 – 432.

[4] LI Q-H, CHEN Y-J, SUN J-G. A New Dictionary Mechanism for Chinese Word Segmentation [J][J]. Journal of Chinese Information Processing, 2003, 4 : 001.

[5] NG H I, LUA K T. A word finding automation for Chinese sentence tokenization[J]. submitted to ACM Transaction of Asian Languages Processing, .

[6] ZHANG H-P, LIU Q, CHENG X-Q, et al. Chinese lexical analysis using hierarchical hidden markov model[C] // Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17. 2003 : 63 – 70.

[7] KUPIEC J. Robust part-of-speech tagging using a hidden Markov model[J]. Computer speech & language, 1992, 6(3) : 225 – 242.

[8] YU H-K, ZHANG H-P, LIU Q, et al. Chinese named entity identification using cascaded hidden Markov model[J]. Journal-China Institute Of Communications, 2006, 27(2) : 87.

[9] MORWAL S, JAHAN N, CHOPRA D. Named entity recognition using hidden Markov model (HMM)[J]. International Journal on Natural Language Computing (IJNLC), 2012, 1(4) : 15 – 23.

[10] MCCALLUM A, FREITAG D, PEREIRA F C. Maximum Entropy Markov Models for Information Extraction and Segmentation.[C] // Icml : Vol 17. 2000 : 591 – 598.

[11] RATNAPARKHI A. A maximum entropy model for part-of-speech tagging[C] // Conference on Empirical Methods in Natural Language Processing. 1996.

[12] BORTHWICK A, GRISHMAN R. A maximum entropy approach to named entity recognition[D]. [S.l.] : Citeseer, 1999.

[13] ZHAO H, HUANG C, LI M. An improved Chinese word segmentation system with conditional random field[C] // Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing. 2006 : 162 – 165.

[14] CONSTANT M, SIGOGNE A. MWU-aware part-of-speech tagging with a CRF model and lexical resources[C] // Proceedings of the workshop on multiword expressions: from parsing and generation to the real world. 2011 : 49 – 56.

[15] EKBAL A, HAQUE R, BANDYOPADHYAY S. Named entity recognition in Bengali: A conditional random field approach[C] // Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II. 2008.

[16] ZHANG Y, CLARK S. Chinese segmentation with a word-based perceptron algorithm[C] // Proceedings of the 45th annual meeting of the association of computational linguistics. 2007 : 840 – 847.

[17] ZHANG Y, CLARK S. Joint word segmentation and POS tagging using a single perceptron[C] // Proceedings of ACL-08: HLT. 2008 : 888 – 896.

[18] COLLOBERT R, WESTON J, BOTTOU L, et al. Natural language processing (almost) from scratch[J]. Journal of machine learning research, 2011, 12(Aug) : 2493 – 2537.

[19] ZHENG X, CHEN H, XU T. Deep learning for Chinese word segmentation and POS tagging[C] // Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013 : 647 – 657.

[20] WU F, LIU J, WU C, et al. Neural Chinese named entity recognition via CNN-LSTM-CRF and joint training with word segmentation[C] // The World Wide Web Conference. 2019 : 3342 – 3348.

[21] CHEN X, QIU X, ZHU C, et al. Long short-term memory neural networks for chinese word segmentation[C] // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015 : 1197 – 1206.

[22] CHEN X, QIU X, ZHU C, et al. Gated recursive neural network for Chinese word segmentation[C] // Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). 2015 : 1744 – 1753.

[23] HATORI J, MATSUZAKI T, MIYAO Y, et al. Incremental joint approach to word segmentation, POS tagging, and dependency parsing in Chinese[C] // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. 2012 : 1045 – 1053.

[24] SHAO Y, HARDMEIER C, TIEDEMANN J, et al. Character-based joint segmentation and POS tagging for Chinese using bidirectional RNN-CRF[J]. arXiv preprint arXiv:1704.01314, 2017.

[25] CONNEAU A, KIELA D, SCHWENK H, et al. Supervised learning of universal sentence representations from natural language inference data[J]. arXiv preprint arXiv:1705.02364, 2017.

[26] MCCANN B, BRADBURY J, XIONG C, et al. Learned in translation: Contextualized word vectors[C] // Advances in Neural Information Processing Systems. 2017 : 6294 – 6305.

[27] SUBRAMANIAN S, TRISCHLER A, BENGIO Y, et al. Learning general purpose distributed sentence representations via large scale multi-task learning[J]. arXiv preprint arXiv:1804.00079, 2018.

[28] RAMACHANDRAN P, LIU P J, LE Q V. Unsupervised pretraining for sequence to sequence learning[J]. arXiv preprint arXiv:1611.02683, 2016.

[29] HOWARD J, RUDER S. Universal language model fine-tuning for text classification[J]. arXiv preprint arXiv:1801.06146, 2018.

[30] CHANG P-C, GALLEY M, MANNING C D. Optimizing Chinese word segmentation for machine translation performance[C]

// Proceedings of the third workshop on statistical machine translation. 2008 : 224 – 232.

[31] TESNIÈRE L. Eléments de syntaxe structurale[J], 1959.

[32] 黄曾阳. HNC 理论概要 [J/OL]. 中文信息学报, 1997, 11(4) : 12. http://jcip.cipsc.org.cn/CN/abstract/article_674.shtml.

[33] DOZAT T, MANNING C D. Deep biaffine attention for neural dependency parsing[J]. arXiv preprint arXiv:1611.01734, 2016.

[34] JI T, WU Y, LAN M. Graph-based dependency parsing with graph neural networks[C] // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019 : 2475 – 2485.

[35] MA X, HOVY E. Neural probabilistic model for non-projective mst parsing[J]. arXiv preprint arXiv:1701.00874, 2017.

[36] FERNÁNDEZ-GONZÁLEZ D, GÓMEZ-RODRÍGUEZ C. Left-to-right dependency parsing with pointer networks[J]. arXiv preprint arXiv:1903.08445, 2019.

[37] KIPERWASSER E, GOLDBERG Y. Simple and accurate dependency parsing using bidirectional LSTM feature representations[J]. Transactions of the Association for Computational Linguistics, 2016, 4 : 313 – 327.

[38] MA X, HU Z, LIU J, et al. Stack-pointer networks for dependency parsing[J]. arXiv preprint arXiv:1805.01087, 2018.

[39] YAROWSKY D. Unsupervised word sense disambiguation rivaling supervised methods[C] // 33rd annual meeting of the association for computational linguistics. 1995 : 189 – 196.

[40] CARRERAS X, MÀRQUEZ L. Introduction to the CoNLL-2005 shared task: Semantic role labeling[C] // Proceedings of the ninth conference on computational natural language learning (CoNLL-2005). 2005 : 152 – 164.

[41] BJÖRKELUND A, HAFDELL L, NUGUES P. Multilingual semantic role labeling[C] // Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task. 2009 : 43 – 48.

[42] BAHDANAU D, CHO K, BENGIO Y. Neural machine translation by jointly learning to align and translate[J]. arXiv preprint arXiv:1409.0473, 2014.

[43] LUONG M-T, PHAM H, MANNING C D. Effective approaches to attention-based neural machine translation[J]. arXiv preprint arXiv:1508.04025, 2015.

[44] HERMANN K M, KOCISKY T, GREFENSTETTE E, et al. Teaching machines to read and comprehend[C] // Advances in neural information processing systems. 2015 : 1693 – 1701.

[45] SERBAN I V, SORDONI A, BENGIO Y, et al. Building end-to-end dialogue systems using generative hierarchical neural network models[C] // Thirtieth AAAI Conference on Artificial Intelligence. 2016.

[46] CHEN H, LIU X, YIN D, et al. A survey on dialogue systems: Recent advances and new frontiers[J]. Acm Sigkdd Explorations Newsletter, 2017, 19(2) : 25 – 35.

[47] LECUN Y, BENGIO Y, OTHERS. Convolutional networks for images, speech, and time series[J]. The handbook of brain theory and neural networks, 1995, 3361(10) : 1995.

[48] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural computation, 1997, 9(8) : 1735 – 1780.

[49] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C] // Advances in neural information processing systems. 2017 : 5998 – 6008.

[50] JOULIN A, GRAVE E, BOJANOWSKI P, et al. Bag of tricks for efficient text classification[J]. arXiv preprint arXiv:1607.01759, 2016.

[51] ZHANG X, ZHAO J, LECUN Y. Character-level convolutional networks for text classification[C] // Advances in neural information processing systems. 2015 : 649 – 657.

[52] TAN J, WAN X, XIAO J. Abstractive document summarization with a graph-based attentional neural model[C] // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017 : 1171 – 1181.

[53] YAO J-G, WAN X, XIAO J. Recent advances in document summarization[J]. Knowledge and Information Systems, 2017, 53(2) : 297 – 336.

[54] BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model[J]. Journal of machine learning research, 2003, 3(Feb) : 1137 – 1155.

[55] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[C] // Advances in neural information processing systems. 2013 : 3111 – 3119.

[56] PENNINGTON J, SOCHER R, MANNING C D. Glove: Global vectors for word representation[C] // Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014 : 1532 – 1543.

[57] PETERS M E, NEUMANN M, IYYER M, et al. Deep contextualized word representations[J]. arXiv preprint arXiv:1802.05365, 2018.

[58] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[C] // Advances in neural information processing systems. 2014 : 3104 – 3112.

[59] SHANG L, LU Z, LI H. Neural responding machine for short-text conversation[J]. arXiv preprint arXiv:1503.02364, 2015.

[60] DEVLIN J, CHANG M-W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[61] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[J]. OpenAI Blog, 2019, 1(8) : 9.

[62] SONG K, TAN X, QIN T, et al. Mass: Masked sequence to sequence pre-training for language generation[J]. arXiv preprint arXiv:1905.02450, 2019.

[63] FENG M, XIANG B, GLASS M R, et al. Applying deep learning to answer selection: A study and an open task[C] // 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). 2015 : 813 – 820.

[64] ZHOU X, LI L, DONG D, et al. Multi-turn response selection for chatbots with deep attention matching network[C] // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018 : 1118 – 1127.

[65] LU J, XIE Z, LING G, et al. Spatio-Temporal Matching Network for Multi-Turn Responses Selection in Retrieval-Based Chatbots[J], .

[66] BLEI D M, NG A Y, JORDAN M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan) : 993 – 1022.