现代汉语词典对“语言”的定义是“人类所特有的用来表达意思、交流思想的工具，是一种特殊的社会现象，由语音、词汇和语法构成一定的系统，一视同仁地为各个阶级服务”。自然语言处理(NLP)是一种基于理论的计算技术，用于人类语言的自动分析和表达。自然语言处理的目的是使计算机处理，理解并能够生成人类的语言，它涉及计算机科学，人工智能和语言学等多个学科的交叉。

自然语言处理的起源可以追溯到1950年图灵提出的“图灵测试”。图灵测试指人类在不知情情况下能否识别于其交谈的对象是否为机器，用来检验计算机是否拥有真正的智能。早期的自然语言处理以基于规则的方法为主流。1956年乔姆斯基借鉴香农的工作，把有限状态机作为工具刻画语法，建立了自然语言的有限状态机模型。基于规则的自然语言处理主

要受到语法的影响，通过分析语言结构，以达到用语法规则约束自然语言处理的目的，其方法的本质是模式匹配。但由于自然语言本身具有的歧义、多样性、以及上下文相关等特点，基于规则的方法在自然语言处理上进展缓慢。然而在20世纪50年代末到60年代中期基于统计的方法开始复苏，多数学者普遍认为只有详尽的历史语料才能带来靠谱的结论。从而诞生了一些比较著名的理论与算法，如贝叶斯方法、隐马尔可夫、最大熵、维特比算法、支持向量机之类。但是总的来说，这个时代依然是基于规则的理性主义的天下，基于统计的经验主义虽然取得了不俗的成就，却依然没有受到太大的重视。

90年代以后基于统计的方法开始大放异彩。有两件事从根本上促进了自然语言处理研究的复苏与发展。一是90年代中期以来，计算机的速度和存储量大幅增加，为自然语言处理改善了物质基础，使得语音和语言处理的商品化开发成为可能；二是1994年Internet商业化和同期网络技术的发展使得基于自然语言的信息检索和信息抽取的需求变得更加突出。研究者采用基于统计的机器学习方法，在语料库基础上建立语言模型，取得了显著的效果。但传统的机器学习算法是基于离散表示的线性模型，不能充分的挖掘语料库信息。

2013年以后word2vec的提出以及神经网络在自然语言处理中的应用，将深度学习与自然语言处理的结合推向了高潮。三种主要类型的神经网络为：循环神经网络（recurrent neural networks）、卷积神经网络（convolutional neural networks）和结构递归神经网络（recursive neural networks）。seq2seq的提出将神经网络分为编码与解码两个部分，完胜了统计机器翻译方法。注意力机制的采用模拟了人类的视觉机制，增强了神经网络的可解释性。直到今天注意力机制仍是一个非常重要的概念。基于自注意力机制的transformer解决了RNN不能并行训练的问题，极大的提高了神经网络的训练速度。2018年bert的提出打破了11项NLP任务的记录，奠定了预训练模型方法的地位。总的来说，面向大量的网络数据资源，基于经验主义的深度学习方法在自然语言处理上取得了丰硕的成果。

随着自然语言处理的进一步发展，目前的高性能系统充分利用了先进的机器学习方法，例如深度学习、强化学习等，还产生了许多高性能工具，例如斯坦福大学开发的语言结构分析器和很多分布式词向量表示工具等。自然语言处理技术快速发展主要取决于以下几个因素：（1）计算能力的增长使得深度神经网络不在停留在理论阶段，同时为更好的算法提供支撑。（2）社交网络的发展产生了大量的文本数据，为提供了高质量的语料库。（3）机器学习算法的快速发展，主要是深度学习的快速发展，产生了很多高效的网络结构。

自然语言处理的领域与技术非常繁多，按照其目的大致可以分为自然语言理解与自然语言生成两种。自然语言理解侧重于如何理解文本，包括的任务有情感分类、实体识别、共指消解、句法分析、关系抽取、阅读理解等。自然语言生成侧重于理解文本后如何生成文本，包含的任务有文本摘要、机器翻译、问答系统、对话系统等。两者间不存在有明显的界限，如阅读理解实际属于问答系统的一个子领域。后面将从机器翻译、问答系统、对话系统这三个领域介绍自然语言处理的应用。

一、机器翻译

机器翻译开始于1933年。苏联科学家彼得·特罗扬斯基向苏联科学院提交了“用于从一种语言翻译到另一种语言时选择和打印单词的机器”。这项发明非常简单——它有四种不同语言的卡片、一台打字机和一台老式胶卷照相机。

操作员从文本中取出第一个单词，找一张对应的卡片，拍摄照片，然后在打字机上键入单词的形态特征(名词、复数、属格)。打字机的按键编码了其中一个特征。磁带和相机的胶卷同时使用，制作了一组带有单词及其形态的框架。

在当时的苏联，这项发明被认为是无用的。直到1956年两位苏联科学家发现了他的专利，世界上才有人知道这台机器。

1954年1月7日，在纽约的IBM总部开始了Georgetown–IBM实验。IBM 701计算机在历史上首次将60个俄语句子自动翻译成英语。而然理想的实验结果掩盖了一个细节——翻译的例子是经过精心挑选和测试的，以排除任何具有歧义的句子或单词。就日常使用而言，这套系统不比一本袖珍常用语手册好多少。但是，这种竞赛依旧开始了:加拿大、德国、法国，尤其是日本，都加入了机器翻译的竞赛。

改进机器翻译的工作持续了四十年之久，但没有取得显著的成果。1966年，美国ALPAC委员会在其著名的报告中称，机器翻译昂贵、不准确、没有前途。相反，他们建议把重点放在词典的发展上，这使得美国的研究人员在近十年的时间里被排除在机器翻译之外。

尽管如此，现代自然语言处理的基础是由科学家和他们的尝试、研究和发展创造的。今天所有的搜索引擎、垃圾邮件过滤器和个人助理的出现都要归功于一些相互监视的国家。

1.1基于规则的机器翻译

围绕基于规则的机器翻译的第一个想法产生在70年代。科学家们仔细观察口译员的工作，试图使用极其迟缓的计算机重复这些动作。这些系统包括:

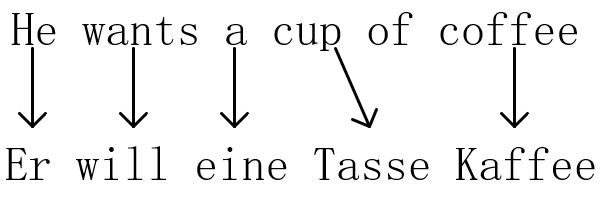
（1）双语词典(俄语 ->英语)

（2）每种语言的一套语言规则(例如，以某些后缀结尾的名词，如-heit， -keit， -ung是阴性的)

仅这两种而已。如果需要，系统可以通过一些修改来补充，比如名字列表、拼写纠正器和音译器等。

1.1.1直接机器翻译

这是最直接的机器翻译类型。 它把文本分成单词，翻译它们，然后纠正形态，并协调语法，使整个句子或多或少听起来正确。富有经验的语言学家为每个单词写下规则。系统的输出返回某种翻译。通常情况下效果糟糕。 语言学家白白浪费了时间。现代系统根本不使用这种方法。

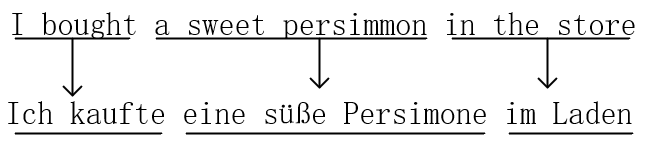


英语->德语的直接翻译

1.1.2 基于转换的机器翻译

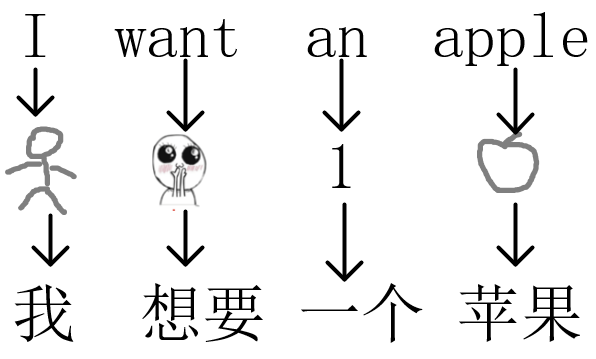
与直接翻译相比，基于转换的方法首先通过确定句子的语法结构来进行翻译，就像我们在学校学习的那样。然后我们操纵整个结构，而不是单词。这有助于在翻译中对词序进行正确的转换。

但是在实践中，它仍然导致了逐字翻译，花费了语言学家大量的精力。一方面，它简化了一般语法规则。但是另一方面，由于与单个单词相比单词构造数量增加，它变得更加复杂。



1.1.3语际机器翻译

通过这种方法，源语言将转换为中间表示形式，并且对于世界上所有语言都是统一的。 这正是笛卡尔梦寐以求的国际语：一种元语言，遵循通用规则并将翻译转换为简单的“来回”任务。国际语可以转换为任何目标语言，这就是国际语的特点。



由于转换的使用，语际翻译经常与基于转换的机器翻译混淆。语际机器翻译的不同之处在于语言规则是针对每种单独的语言和中间语言的，而不是针对语言对。这意味着，我们可以在语际系统中添加第三种语言，并在这三种语言之间进行翻译。在基于转换的系统中，我们无法做到这一点。

它看起来很完美，但在实际中却并非如此。创建这样的通用语言非常困难，许多科学家一生都在为之努力。他们并没有成功，但是由于有了他们的工作，我们现在有了形态，句法甚至语义层次的表示形式。

基于人工书写翻译规则的机器翻译方法主要优点是直观，语言学家可以非常容易将翻译知识利用规则的方法表达出来，书写的翻译规则可读性比较好。其次，翻译规则的书写颗粒度具有很大的可伸缩性。较大颗粒度的翻译规则很强的概括能力，比如通用翻译规则；较小颗粒度的翻译规则具有精细的描述能力，比如个性翻译规则。第三，翻译规则便于处理复杂的句法结构和进行深层次的语义理解，比如解决翻译过程中的长距离依赖问题。第四，基于规则的机器翻译系统适应性强，完全不依赖于具体的双语训练语料，这一点不同于基于实例的机器学习方法，包括后来的统计机器翻译和神经机器翻译方法。

基于规则的机器翻译方法最大问题在于人工书写翻译规则的难度挺大，代价非常高，这一点也是被大家所诟病最多的，后来的基于实例的机器翻译方法主要攻击的也是这一点，认为机器学习方法可以脱离人工书写翻译规则的苦海。第二，人工书写翻译规则的主管因素重，因人而异，有时与客观事实有一定差距。第三，翻译规则的覆盖性差，特别是细颗粒度的翻译规则很难总结比较全面，比如英语中的所有不规则动词，德语中的可分离前缀。第四，前面提到翻译规则通常按照形式文法规范来书写的，有些复杂的语言现象难以描述。第五，翻译规则通常不具有优先级，系统调试非常枯燥乏味，并且新增加的翻译规则容易与之前存在翻译规则发生冲突，称之为跷跷板现象，这个问题目前还没有很好的解决方法。

从实际应用的角度来看，基于规则的机器翻译方法鲁棒性不够好，经常会出现一些复杂句子或者不太规范的句子翻译失败的现象，本质上是找不到合适的翻译规则来完成整个翻译过程，简单来说就是容错能力较差。

1.2基于实例的机器翻译

日本对机器翻译特别感兴趣。原因是该国很少有人会英语。在即将到来的全球化聚会上，这肯定是一个大问题。因此，日本人非常想找到一种机器翻译的工作方法。

而然基于规则的英日翻译极为复杂。语言结构完全不同，几乎所有单词都必须重新排列并添加新单词。1984年，京都大学的长冈诚（Makoto Nagao）提出了使用现成短语替代重复翻译的想法。

假设我们必须翻译一个简单的句子-“我要去电影院。”假设我们已经翻译了另一个类似的句子-“我要去剧院了”-我们可以在字典中找到“电影”一词。我们所需要做的就是弄清楚两个句子之间的区别，翻译丢失的单词，然后不要将其弄乱。我们的例子越多，翻译效果越好。

基于实例的机器翻译向来自世界各地的科学家展示了一个曙光：事实证明，您只需为计算机提供现有的翻译，而无需花费时间制定规则和特例。这还称不上一个是革命，但显然是迈出了第一步。统计机器翻译的革命性发展将在未来短短五年内发生。

1.3统计机器翻译

1990年初，IBM研究中心首次展示了一个机器翻译系统，它对规则和语言学一无所知。它分析了两种语言的相似文本，并试图理解其中的模式。这个想法简单而美妙。将两种语言中相同的句子分解成单词，然后再进行匹配。这项操作重复了大约5亿次，以计算“Das Haus”以此翻译成“house”、“building”、“construction”的次数，等等。

如果大部分情况下源词被翻译成“house”，机器就会使用它。请注意，我们没有制定任何规则，也没有使用任何词典——所有的结论都是由机器完成的，由统计数据和“如果人们这样翻译，我也会这样翻译”的逻辑来完成。统计机器翻译（SMT）由此诞生。

该方法比所有以前的方法都更加有效和准确。 不需要语言学家。 我们使用的文本越多，翻译的效果就越好。

还剩下一个问题：机器如何将“ Das Haus”一词与“ building”一词相关联？我们如何知道这些是正确的翻译？

答案是我们不知道。开始时，机器假定单词“ Das Haus”与翻译句子中的任何单词均等相关。 接下来，当“ Das Haus”出现在其他句子中时，与“ house”的相关数将增加。 这就是“单词对齐算法”，这是一个机器学习的典型任务。

1.4神经机器翻译

2013年，Nal Kalchbrenner和 Phil Blunsom提出了一种用于机器翻译的新型端到端编码器-解码器架构。该模型使用卷积神经网络（CNN）将给定的源文本编码为连续向量，然后使用循环神经网络（RNN）作为解码器将状态向量转换为目标语言。他们的研究可视为神经机器翻译（NMT）的开端，NMT是一种使用深度学习神经网络在自然语言之间进行映射的方法。NMT的非线性映射不同于线性SMT模型，NMT使用连接编码器和解码器的状态向量来描述语义等价。从另一个角度来理解——一个编码器只能将句子编码为一组特定的特征（状态向量），而另一个神经网络只能将其解码回文本。彼此都只知道自己的语言。这是另一种形式的国际语！

然而，梯度消失/爆炸问题使得RNN实际上很难处理长距离依赖的问题；对应地，NMT模型起初也不能实现很好的性能。Seq2seq模型使用RNN的变体LSTM作为编码器与解码器，由于LSTM中的门机制允许状态向量更新和遗忘，能够更好的捕捉长距离依赖，梯度消失/爆炸问题得以控制。注意力机制模拟了人类视觉专注局部的机制，当解码器解码时也应关注于当前输出最相关的单词部分。之后，NMT的性能得到了显著的提升，带注意力的seq2seq模型成为NMT最先进的模型。

2017年由谷歌提出的transformer模型解决了RNN模型不能并行训练的问题，极大的提高了神经网络的训练速度，并且谷歌在论文中强调了注意力的重要性。

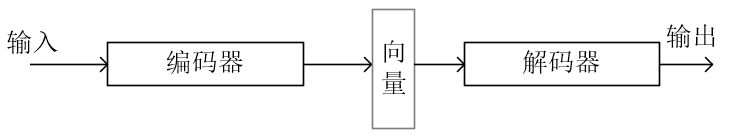
在两年内，神经网络超越了过去20年翻译中出现的一切。神经翻译减少了50％的单词顺序错误，17％的词汇错误和19％的语法错误。神经网络甚至学会了用不同语言协调性别和大小写。而且没有人教他们这样做。

最明显的改进发生在从未使用直接翻译的领域。统计机器翻译方法始终以英语为主要来源。因此，如果您从俄语翻译为德语，则机器首先将文本翻译为英语，然后再从英语翻译为德语，这会造成双重损失。

神经翻译不需要——只需解码器即可工作。这是首次没有普通词典的语言之间的直接翻译成为可能。

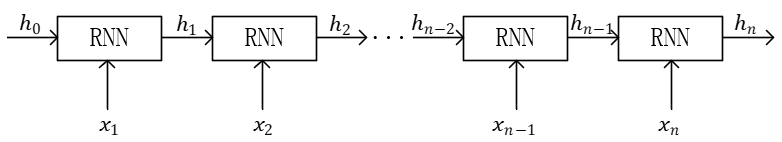
1.5 seq2seq模型

seq2seq是一个编码器-解码器结构的模型，如下图所示，编码器的输入是一个可变长度的序列，它将输入文本编码为固定长度的上下文向量。理想情况下，这个向量中包含了输入序列中的所有信息。然后解码器对这个向量进行解码，输出一个可变长度的序列。



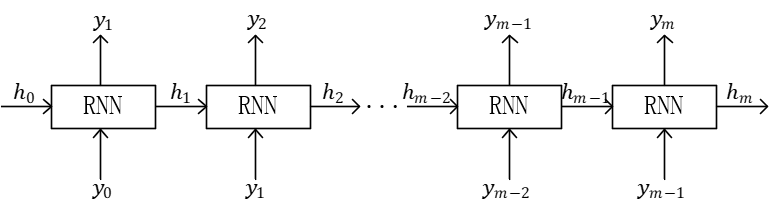
编码器

由于输入输出为可变长度的序列，因此多数神经网络使用RNN作为编码器与解码器。下图展示了一个基于RNN的编码器结构。其中为我们的输入序列,表示句子中的第个文字。在时刻，我们将输入RNN，得到其输出,之后重复这一步骤直到所有的输入序列全部输入到RNN中，得到最终的编码后的向量。这就是一个编码器的主要流程。在时，初始化为0。



基于RNN的编码器结构

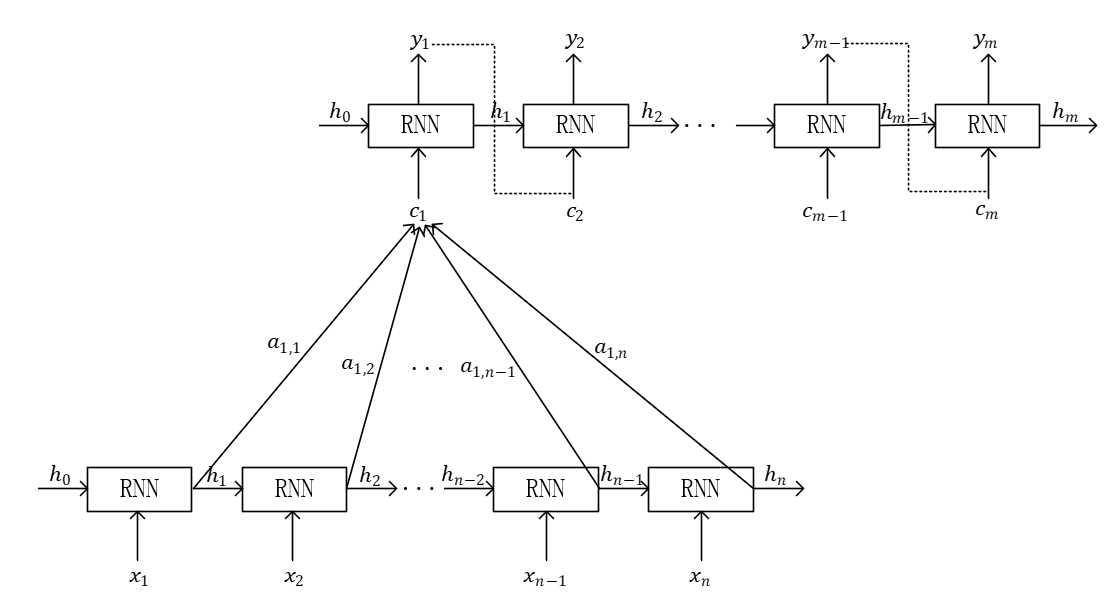
下图展示了一个基于RNN的解码器结构。其中为我们的输出序列。解码器将编码器输出的上下文向量和(初始化为0)做为初始的输入。类似于编码器，在时刻，我们将输入RNN，得到其输出,之后重复这一步骤直到解码器输出序列的终止符。

 `

1.5.1注意力机制

注意力机制是编码器-解码器结构的神经网络一个非常重要的概念。举个例子，当你翻译“I want an apple”这样一个非常简单的句子的时候，你的翻译结果的第一个字“我”是仅仅与第一个单词“I”相关的。但在上面的seq2seq模型中，解码器需要从编码器的最终输出向量中解码出第一个单词。如果输入序列很长时，单词自身的信息已经丢失，解码器不能解码。因此，我们需要引入注意力机制。

下图展示了带有注意力机制的seq2seq模型结构。其中编码器保持不变，在解码器中我们使用来表示上下文向量。为编码器中每一个上下文向量的加权之和。我们为每一个赋予不同的权重（图中的），从而达到注意力放在局部某个单词的目的。



2、问答系统

智能问答(Question Answer, QA)系统就是基于大量语料数据，通过数学模型，通过相关编程语言实现的一个能够和人类进行对话，解决问题的一个软件系统。自然语言问题大致可以分为七类，事实类(factoid)问题、是非类(yes/no)问题、定义类(definition)问题、列表类(list)问题，比较类(comparison)问题、意见类(opinion)问题和指导类(how-to)问题。

1. 事实类问题对应的答案是现实世界中的一个或多个实体

2. 是非类问题对应的答案只能是“是”或者“否”。

3. 定义类问题对应的答案是关于问题中提到的某个实体的术语解释。

4. 列表类问题对应的答案通常是一个集合。

5. 比较类、意见类和指导类问题对应的答案通常较为主观。

问答系统的发展可以追溯到图灵测试，其可以看作是问答系统的一个蓝图。早期的两个比较著名的QA系统：BASEBALL（1961年）和LUNAR（1973年）。BASEBALL可用来回答美国一个季度棒球比赛的时间地点成绩等 自然语言问题。LUNAR可帮助地质学家方便的了解、比较和苹果股阿波罗登月计划积累的月球突然和岩石的各种化学分析数据。这个时期主要是限定领域、处理结构数据的问答系统。在用户提问时，系统把用户的问题转换成SQL查询语句，从数据库中查询到数据提供给用户。

20世纪 90年代, 问答系统进入开放领域、 基于文本的新时期。由于互联网的飞速发展, 产生了大量的电子文档, 这为问答系统进入开放领域、 基于文本的时期提供了客观条件。特别是在1999 年 TREC(text retrieval conference)的 QA track设立以来, 极大地推动了问答系统的发展。

问答系统可以分为三个流程1、问题分析：分析问题，理解问题。2、信息检索：根据问题的分析结果去缩小答案 可能存在的范围。3、答案抽取：从可能存在答案的信息块中抽取答案。根据答案的来源可以将问答系统分为：基于知识的问答系统、基于信息检索的问答系统、基于问答对的问答系统.

2.1基于知识图谱的问答系统

基于知识图谱的问答通过将自然语言问题映射到结构化数据库上的查询来回答自然语言问题的想法。就像基于文本的问题解答范式一样，这种方法可以追溯到自然语言处理的最早时期，其基础是诸如BASEBALL之类的系统，它从棒球和统计数据的结构化数据库中提取答案。从文本字符串映射到任何逻辑形式的表示称为语义解析器。用于问答系统的语义解析器通常映射到谓词演算或查询语言。例如将”谁是阿里巴巴的创始人?”, 解析为：创建（?X，阿里巴巴）。

将问题解析为机器的查询语言后，我们就可以直接查询知识图谱获取答案。具体的流程如下：

（1）将问题的词语或单词与知识库中的实体或者关系进行映射，这些映射构成了一棵树的叶子节点。

（2）对叶子节点进行链接、求交和聚合三种操作自下向上构建多棵语法树。

（3）使用一个预先训练好的机器学习模型将正确的语法树区分出来，最终语法树的根节点则为输出的查询语句。

为了区分出正确的语法树，需要大量的带有逻辑表达式的自然语言标注数据，从而训练得到机器学习模型。但这仍只能满足有限的逻辑逻辑表达式，使用场景有限。

而另一种方法将基于知识图谱问答看做一个语义匹配过程。通过表示学习知识图谱以及问题的语义表示，得到低维空间的嵌入向量，再通过数值计算，直接匹配与用户问句语义最相似的答案。即问答任务就可以看成问句语义向量与知识谱图中实体、边的语义向量相似度计算的过程。具体的流程如下：

（1）问题编码：问题使用神经网络（通常为Bi-LSTM）进行编码，得到问题的低维嵌入向量。

（2）答案编码：答案不能直接映射成词向量，一般是将答案分为四块编码。利用到答案实体，答案类型，答案关系，答案上下文信息。

（3）向量匹配：使用答案编码分别和问句向量做相似度计算，最终的相似度为几种相似度之和。

这种基于向量匹配的虽然不需要大量的自然语言标注数据，系统构建难度小。但其可解释较差，并且在复杂问题上表现不好。

2.2基于信息检索的问答系统

基于信息检索的问答的目的是通过在网络上或其他文档集合中查找简短的文本段来回答用户的问题。基于信息检索的问答系统的基础是机器阅读理解，利用机器阅读理解技术进行问答即是对非结构化文章进行阅读理解得到答案，可以分成匹配式问答，抽取式问答和生成式问答，目前绝大部分是抽取式问答。下面介绍抽取式问答的步骤。

2.2.1 问题处理

问题处理阶段的主要目标是提取问题将问题形式化，传递给信息检索系统用来匹配潜在文档的关键字。一些系统还额外提取一些更多的信息，比如：答案类型、问题类型、焦点（问题中可能被答案所替代的单词）。对问题“哪个省份的省会人口最多？”有如下信息：

答案类型：城市

问题：有最多人口的省会

焦点：省会

2.2.2信息检索

问题处理阶段生成的查询被发送到信息检索引擎，从而产生一组文档，这些文档按与查询问题的相关性进行排序。因为大多数答案提取方法都被设计用于较小的区域（例如段落），所以问答系统接下来将前N个文档分成较小的段落（例如章节、段落或句子）。这些可能已经在源文档中进行了细分，或者我们可能需要运行段落细分的算法。

然后，段落检索的最简单形式是简单地将每个段落传递到答案提取阶段。一个更复杂的变体是通过对检索到的段落运行命名实体或答案类型分类来过滤段落，并丢弃不包含问题答案类型的段落。还可以使用监督学习通过以下特征对其余段落进行排名：

段落中正确类型的命名实体的数量

段落中出现的问题关键词的数量

段落中出现问题关键字的最长序列

段落所在的文档的排名

关键字与原始问题之间的相似度

2.2.3 答案抽取

问答系统的最后一步是提取文章中的具体答案，比如回答“珠穆朗玛峰有多高?”这样的问题。这个任务通常是通过标记片段来完成的:给定一篇文章，识别组成答案的片段。

答案抽取的一个简单的基线算法是对候选文章运行一个命名实体识别器，并返回文章中正确答案类型的片段。因此，在下面的例子中，将从文章中提取划线命名的实体，作为对距离/数量问题的答案:

珠穆朗玛峰有多高？

珠穆朗玛峰的高度是8848米。

而然许多问题（例如定义类问题）的答案往往不是特定的命名实体类型。因此，有关答案抽取的工作通常使用基于监督学习的更复杂的算法。本节将介绍一个基于神经网络的答案抽取算法。

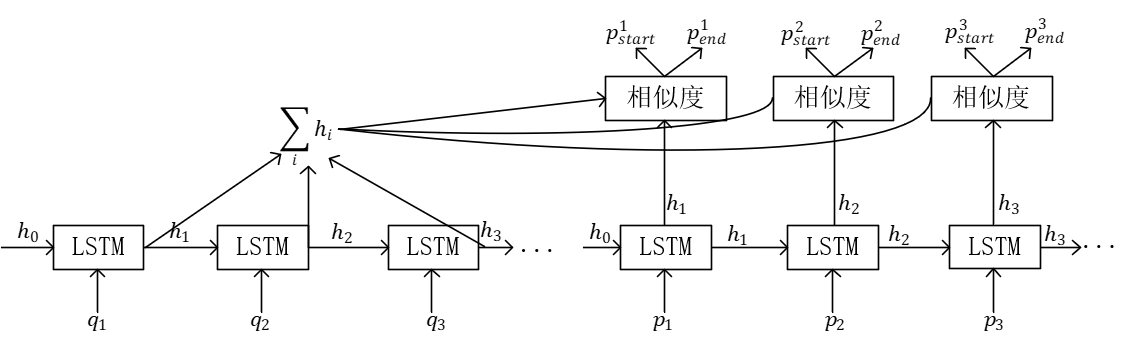
**神经网络的答案抽取**

神经网络的答案抽取方法基于这样的直觉：问题及其答案在语义上以某种方式相似。它具体表现为，通过计算问题的低维嵌入向量和段落的每个片段的低维嵌入向量，然后选择其嵌入向量最接近问题嵌入向量的段落片段作为答案。

神经网络方法的答案抽取通常是在阅读理解任务的背景下设计的。阅读理解任务是自然语言理解的任务之一，也被作为问答系统中的阅读组件。

基于神经网络的机器阅读理解模型是给定问题和段落。其输出是计算对于每个段落中的，它作为一个答案片段的开头的概率，和它作为一个答案片段的结尾的概率。可以看到这是一个序列到序列的问题，因此同意可以使用编码器-解码器结构的神经网络结构。

下面展示了一个使用RNN的变体LSTM的答案抽取模型。网络左边对问题进行编码，然后将每一步的上下文向量加权求和，得到问题的嵌入向量。右边对可能包含问题的段落进行编码，对于每个通过问题的嵌入向量，模型通过相似度计算其作为答案开始和结束的概率。



2.3基于问答对的问答系统

随着互联网技术的成熟和普及, 网络上出现了常问问题(frequent asked questions, FAQ)数据, 特别是在 2005 年末以来大量 的社区问答(community based question answering, CQA)数据(例如 Yahoo！Answer)出现在网络上, 即有了大量的问题答案对数据, 问答系统进入了开放领域、基于问题-答案对时期。

FAQ问答系统通常有两种实现：1）相似问题匹配，即计算用户问题与现有知识库中的问题的相似度，返回用户问题对应的最精准的答案；2）问题答案匹配，即计算用户问题与知识库中答案的匹配度，返回用户问题对应的最精准的答案，该方案是选择答案，及QA匹配。在这两种方案中都需要用到文本匹配计算。

3.对话系统与聊天机器人

对话系统按照其功能可以分为任务导向型与非任务导向型（也称聊天机器人）。面向任务的对话系统使用与用户之间的对话来帮助完成任务，比如，控制设备、查找餐厅、拨打电话。而聊天机器人专注于在开放的领域与人交谈。旨在模仿非结构化对话或人与人真实的聊天特征，主要用于娱乐，但也用于使面向任务的对话系统变得更自然。在本节中，我们将讨论三种主要的聊天机器人架构：基于规则的系统，信息检索系统和生成式模型。

3.1基于规则的聊天机器人

ELIZA是一个完全基于规则的聊天机器人，它模拟了一个心理医生 Carl Rogers。ELIZA 诞生于 MIT 人工智能实验室，历时三年（1964 - 1966）开发。

ELIZA 预定义好了很多模式（Pattern），每个模式都有其对应的转换方法（Transform）来生成答复，一个 Pattern + Transform 可以理解为一条规则。当对话遇到对应的模式时，ELIZA 会转换然后生成答复。举个例子：

(0 YOU 0 ME) [pattern]

(1 2 3 4) [index]

->

(WHAT MAKES YOU THINK I 3 YOU) [transform]

Transform 里的数字对应 index 位置的用户单词。按照这个规则，一轮对话如下：

You hate me [用户]

->

WHAT MAKES YOU THINK I HATE YOU [聊天机器人]

ELIZA 给每条规则对应了一个关键词，不同关键字有不同的权重，因此，当一句话出现了多个关键词匹配时，优先使用权重高的规则。例如，“I know everybody laughed at me” 这句话，有 “I” 和 “everybody” 两个关键词，但是前者更常见，后者更少见，权重更高，因此 ELIZA 会匹配与 “everybody” 相关的规则。

基于规则的聊天机器人需要大量的时间去构建规则，并且模型不够灵活，不能够很好的联系对话的上下文信息。因此随着互联网的发展，大量文本语料的产生，基于数据的聊天机器人逐渐成为主流，主要包括检索型与生成型。

3.2基于检索的聊天机器人

基于检索的方法从候选回复中选择回复。准确来说，检索式模型的输入是一段上下文内容，和一个可能作为回复的候选答案，模型的输出是对这个候选答案的打分。寻找最合适的回复内容的过程是：先对一堆候选答案进行打分及排序，最后选出分值最高的那个作为最终回复。检索方法的关键是询问-回复匹配。直觉来说，一轮对话中的询问-回复中语义上相近的词越多，那询问-回复越可能是一对正确的对话。这样的假设确实有一定的道理，但事实上询问-回复并不一定是语义上的相近，有时候q询问-回复在语义向量上并没有什么相似性。匹配算法必须克服询问和回复之间的语义鸿沟。

模型选择一个自然的、与整个上下文相关的回复。重要的是要在之前的话语中找出重要的信息，并恰当地模仿话语的关系，以确保谈话的连贯性。

多轮对话的难点在于不仅要考虑当前的问题，也要考虑前几轮的对话情景。多轮对话的难点主要有两点：

1.如何明确上下文的关键信息（关键词，关键短语或关键句）；

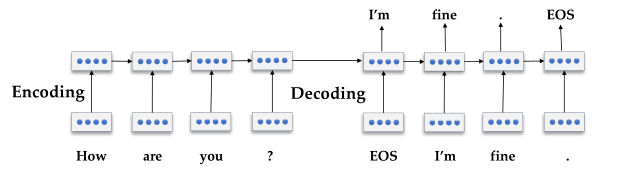
2.在上下文中如何模拟多轮对话间的关系。

现有检索模型的缺陷：在上下文中容易丢失重要信息，因为它们首先将整个上下文表示为向量，然后将该上下文向量与响应sentence向量进行匹配。

3.3基于生成的聊天机器人

深度学习在机器翻译中的成功应用，即神经机器翻译，激发了人们对神经生成对话研究的兴趣。聊天机器人的生成时模型可以看成机器翻译的序列标注问题，即将问句编码之后得到嵌入向量，将这个向量作为解码器的输入，最终返回对问句的回复。

编码器-解码器模型的优点之一是可以更好的考虑对话上下文。使对话保持连贯和增进用户体验。在神经网络中可以输如代表整个对话历史(包括当前的信息)，用连续的表示或嵌入单词和短语来解决上下文敏感的对话生成这一难题。



编码器-解码器结构的对话生成模型

在当前Seq2Seq对话系统中，一个具有挑战性的问题是，它们倾向于产生无关紧要的或不明确的、普通的、几乎没有意义的回复，而这些回复常常涉及到一些诸如“I don't know”，“I am OK”这样的无意义回复。解决这类问题的一个很有效的方法是找到并设置一个更好的目标函数。除此之外，解决此类问题的一个方法是增加模型的复杂度。