打印该邮件 关闭该邮件

【分享文章】自然语言处理之机器智能

"马晓雨" <malittlerain@126.com>

收件人：sheng4444@163.com

时　间：2012-11-8 17:33:37

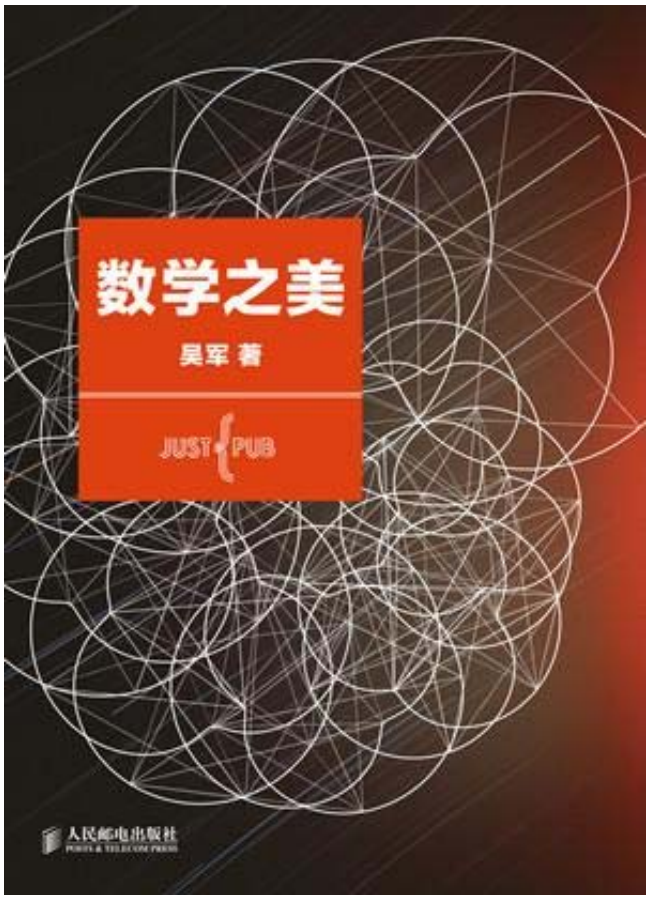
附　件：

这是我在网易云阅读《程序员》上看到的文章，有嚼头。网易云阅读内容很给力，强烈

自然语言处理之机器智能

2012-06-01 16:25:47

语言出现的目的是为了人类之间的通信，字母（或者中文的笔画）、文字和数字实际上是信息编码的不同单位。任何一种语言都是一种编码的方式，而语言的语法规则是编解码的算法。我们把一个要表达的意思，通过某种语言的一句话表达出来，就是用这种语言的编码方式对头脑中的信息做了一次编码，编码的结果就是一串文字。而如果对方懂得这门语言，他或者她就可以用这门语言的解码方法获得说话人要表达的信息。这就是语言的数学本质。虽然传递信息是动物也能做到的，但是利用语言来传递信息是人类的特质。



1946 年，现代电子计算机出现以后，计算机在很多事情上做得比人还好。既然如此，机器是否能够懂得自然语言呢？事实上当计算机一出现，人类就开始琢磨这件事。这里面涉及到两个认知方面的问题：第一，计算机是否能处理自然语言；第二，如果能，那么它处理自然语言的方法是否和人类一样。这本书将回答这两个问题。为了不吊读者的胃口，我在这里先给出简洁版的答案：对这两个问题的回答都是肯定的，Yes ！

最早提出机器智能设想的是计算机科学之父阿兰· 图灵（Alan Turing），1950 年他在《思想》（Mind）杂志上发表了一篇题为“计算的机器和智能”的论文。在论文中，图灵并没有提出什么研究的方法，而是提出了一种来验证机器是否有智能的

方法：让人和机器进行交流（图1），如果人无法判断自己交流的对象是人还是机器时，就说明这个机器有智能了。这种方法被后人称为图灵测试（Turing Test）。图灵其实是留下了一个问题，而非答案，但是一般认为自然语言的机器处理（现在称作自然语言处理）的历史可以追溯到那个时候，至今已经60多年了。

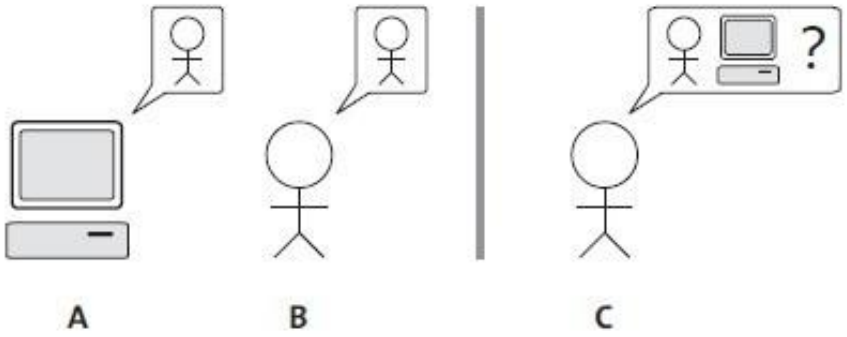


图1 搞不清后面是人还是机器

自然语言处理60 多年的发展过程，基本上可以分成两个阶段。早期的20多年，即从20 世纪50 年代到70 年代，是科学家们走弯路的阶段。全世界的科学家对计算机处理自然语言的认识都被自己局限在人类学习语言的方式上，也就是说，用电脑模拟人脑，这20 多年的成果近乎为零。直到20 世纪70 年代，一些自然语言处理的先驱重新认识这个问题，找到了基于数学模型和统计的方法，自然语言处理进入第二个阶段。30 多年来，这个领域取得了实质性的突破，并且让自然语言处理在很多产品中得以广泛应用。虽然早期自然语言处理的工作对今天没有任何指导意义，但是了解几代科学家的认识过程，对我们了解自然语言处理的方法很有好处，同时也让我们避免走前人的弯路。

让我们回到1956 年的夏天。28 岁的约翰·麦卡锡（John McCarthy），以及同年龄的马文·明斯基（Marvin Minsky），37 岁的罗切斯特（Nathaniel Rochester）和40 岁的香农，他们4 人提议在麦卡锡工作的达特茅斯学院2 开了一个被他们称为“达特茅斯夏季人工智能研究会议”的头脑风暴式的研讨会。参加会议的还有6 位年轻的科学家，包括40 岁的赫伯特•西蒙（Herbert Simon）和28 岁的艾伦·纽维尔（Allen Newell）。在这次研讨会上，这10 个人讨论当时计算机科学尚未解决的问题，包括人工智能、自然语言处理和神经网络等。人工智能这个提法便是在这次会议上提出的。这10 个人，除了香农，当时大多没有什么名气。但是没关系，这些年轻人默默无闻的时间不会太久，后来这些人都非常了不起，其中出了4 位图灵奖获得者（麦卡锡、明斯基、西蒙和纽维尔）。当然香农不必得什么图灵奖，作为信息论的发明人，他在科学史上的地位和图灵是相当的，而且通信领域的最高奖就是以他的名字命名的。

达特茅斯会议的意义超过10 个图灵奖。这10 位后来被证明是20 世纪IT领域最优秀的科学家，开创了很多今天依然活跃的研究领域，而这些研究领域的成功使我们的生活变得十分美好。遗憾的是，受历史的局限，这10 个世界上最聪明的头脑一个月的火花碰撞，并没有产生什么了不起的思想，他们对自然语言处理理解的总和，甚至不如今天一位世界一流大学的博士毕业生。这是因为在当时，全世界对自然语言处理的研究都陷入了一个误区。

那时候学术界对人工智能和自然语言理解的普遍认识是这样的：要让机器完成翻译或者语音识别这样只有人类才能做的事情，就必须先让计算机理解自然语言，而做到这一点就必须让计算机有类似我们人类这样的智能。（今天几乎所有的科学家都不坚持这一点，而很多的门外汉还误以为计算机是靠类似我们人类的这种智能解决了上述问题。）为什么会有这样的认识？因为人类就是这么做的，道理就这么简单。对于人类来讲，一个能把英语翻译成汉语的人，一定是能非常好地理解这两种语言的。这就是直觉的作用。在人工智能领域，包括自然语言处理领域，后来把这样的方法论称作“鸟飞派”，也就是看看鸟是怎样飞的，就能模仿鸟造出飞机，而不需要了解空气动力学。事实上我们知道，怀特兄弟发明飞机靠的是空气动力学而不是仿生学。在这里，我们不要笑话我们前辈来自于直觉的天真想法，这是人类认识的普遍规律。今天，机器翻译和语音识别已经做得不错，并且有上亿人使用过，但是大部分这个领域之外的人依然错误地以为这两个应用是靠计算机理解了自然语言而完成的。事实上，它们全都靠得是数学，更准确地说是靠统计。

在20世纪60年代，摆在科学家面前的问题是怎样才能理解自然语言。当时普遍的认识是首先要做好两件事，即分析语句和获取语义。这实际上又是惯性思维的结果 — 它受到传统语言学研究的影响。从中世纪以来，语法一直是欧洲大学教授的主要课程之一。到16世纪，伴随着《圣经》被翻译介绍到欧洲以外的国家，这些国家的语言语法逐步得到完善。到18、19 世纪，西方的语言学家们已经对各种自然语言进行了非常形式化的总结，这方面的论文非常多，形成了十分完备的体系。学习西方语言，都要学习它们的语法规（Grammar Rules）、词性（Part of Speech）和构词法（Morphologic）等。当然，应该承认这些规则是我们人类学习语言（尤其是外语）的好工具。而恰恰这些语法规则又很容易用计算机的算法描述，这就更坚定了大家对基于规则的自然语言处理的信心。

对于语义的研究和分析，相比较而言要不系统得多。语义也比语法更难在计算机中表达出来，因此直到20 世纪70 年代，这方面的工作仍然乏善可陈。值得一提的是，中国古代语言学的研究主要集中在语义而非语法上。很多古老的专著，比如《说文解字》等都是语义学研究的成果。由于语义对于我们理解自然语言是不可或缺的，因此各国政府在把很大比例的研究经费

提供给“句法分析”相关研究的同时，也把一部分钱给了语义分析和知识表示等课题。现在把当时科学家头脑里的自然语言处理从研究到应用的依赖关系用图2来描述。

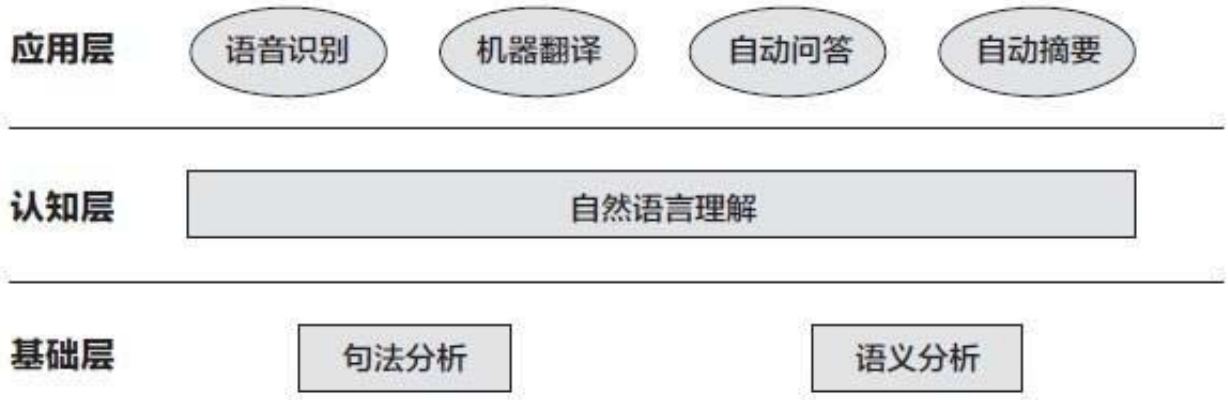


图2 早期对自然语言处理的理解
让我们集中看看句法分析。先看下面一个简单的句子

徐志摩喜欢林徽因。

这个句子可以分为主语、动词短语（即谓语）和句号三部分，然后可以对每个部分作进一步分析，得到如下的语法分析树（Parse Tree）：

图3 句子的语法分析树
分析它采用的文法规则通常被计算机科学家和语言学家称为重写规则（Rewrite Rules），
具体到上面的句子，重写规则包括：

句子→主语谓语句号
主语→名词
谓语→动词 名词短语
名词短语→名词
名词→徐志摩
动词→喜欢
名词→林徽因
句号→。

20 世纪80 年代以前，自然语言处理工作中的文法规则都是人写的，这和后来采用机器总结的做法大不相同。直到2000 年后，很多公司，比如著名的机器翻译公司SysTran，还是靠人来总结文法规则。20 世纪60 年代，基于乔姆斯基形式语言的编译器技术得到了很大的发展，计算机高级程序语言都可以概括成上下文无关的文法，这是一个在算法上可以在多项式时间内解决的问题（Polynomial Problem，详见附录）。高级程序语言的规则和上述自然语言的规则从形式上看很相似。因

此，很容易想到用类似的方法分析自然语言。那时，科学家们设计了一些非常简单的自然语句的语法分析器（Parser），可以分析词汇表为百十来个词、同时长度为个位数的简单语句（不能有很复杂的从句）。
科学家们原本以为随着对自然语言语法概括得越来越全面，同时计算机计算能力的提高，这种方法可以逐步解决自然语言理解的问题。但是这种想法很快遇到了麻烦。我们从上图可以看出，句法分析实际上是一件很罗嗦的事：一个短短的句子居然分析出这么一个复杂的二维树结构，而且居然需要八条文法规则，即使刨去词性标注的后四条依然还有四条。
当然让计算机处理上述分析还是不难的，但要处理下面《华尔街日报》的一个真实句子，就不是那么容易办到了：

> 美联储主席本·伯南克昨天告诉媒体7 000 亿美元的救助资金将借给上百家银行、保险公司和汽车公司。

虽然这个句子依然符合“句子→主语谓语句号”这条文法规则：

> 主语【美联储主席本·伯南克】|| 动词短语【昨天告诉媒体7 000 亿美元的救助资金将借给上百家银行、保险公司和汽车公司】|| 句号【。】

然后，接下来可以进行进一步的划分，比如主语“美联储主席本·伯南克”分解成两个名词短语“美联储主席”和“本·伯南克”，当然，前者修饰后者。对于动词短语也可以做同样的分析。这样，任何一个线性的语句，可以被分析成这样一棵二维的语法分析树（Parse Tree）。我们没有将完整的分析树画出来，是因为在这本书一页纸上，无法画出整个语法分析树 — 这棵树非常大，非常复杂。应该讲，单纯基于文法规则的分析器是处理不了上面这样复杂的语句的。
这里面至少有两个越不过去的坎儿。首先，要想通过文法规则覆盖哪怕20% 的真实语句，文法规则的数量（不包括词性标注的规则）至少是几万条。语言学家几乎已经是来不及写了，而且这些文法规则写到后来甚至会出现矛盾，为了解决这些矛盾，还要说明各个规则特定的使用环境。如果想要覆盖50% 以上的语句，文法规则的数量最后会多到每增加一个新句子，就要加入一些新的文法。这种现象不仅出现在计算机处理语言上，而且出现在人类学习和自己母语不同语系的外语时。今天30 岁以上的人都应该会有这种体会：无论在中学和大学英语考试成绩多么好，也未必能考好GRE，更谈不上看懂英文的电影。原因就是我们即使学了10 年的英语语法，也不能涵盖全部的英语。
其次，即使能够写出涵盖所有自然语言现象的语法规则集合，用计算机解析它也是相当困难的。描述自然语言的文法和计算机高级程序语言的文法不同。自然语言在演变过程中，产生了词义和上下文相关的特性。因此，它的文法是比较复杂的上下文有关文法（Context Dependent Grammar），而程序语言是我们人为设计的，为了便于计算机解码的上下文无关文法（Context Independent Grammar），相比自然语言而言简单得多。理解两者的计算量不可同日而语。
在计算机科学中，图灵奖得主高德纳（Donald Knuth）提出了用计算复杂度（Computational Complexity）来衡量算法的耗时。对于上下文无关文法，算法的复杂度基本上是语句长度的二次方，而对于上下文有关文法，计算复杂度基本上是语句长度的六次方。也就是说，长度同为10 的程序语言的语句和自然语言的语句，计算机对它们进行语法分析的计算量，后者是前者的一万倍。而且随着句子长度的增长，二者计算时间的差异会以非常快的速度扩大。即使今天，有了很快的计算机（英特尔i5 双核处理器），分析上面这个二三十个词的句子也需要几分钟的时间。因此，在20 世纪70 年代，即使是制造大型机的IBM 公司，也不可能采用规则的方法分析一些真实的语句。
本文节选自《数学之美》一书，吴军著，由人民邮电出版社出版。