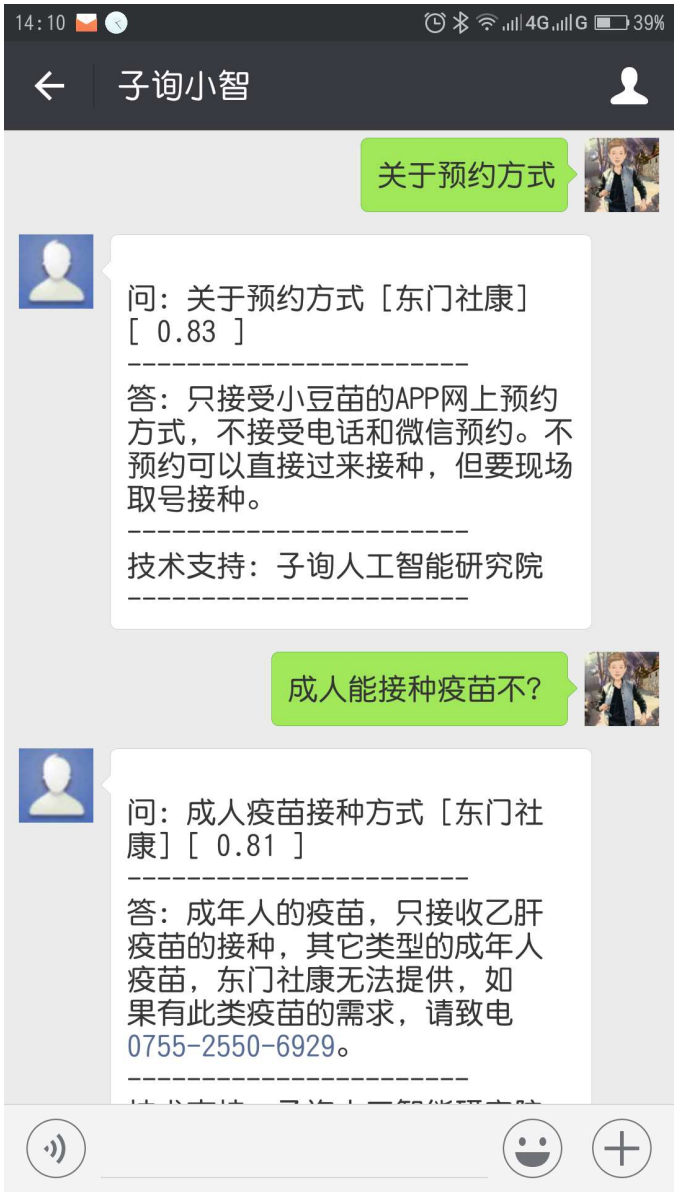
人工智能在自然语言处理中的应用初步研究

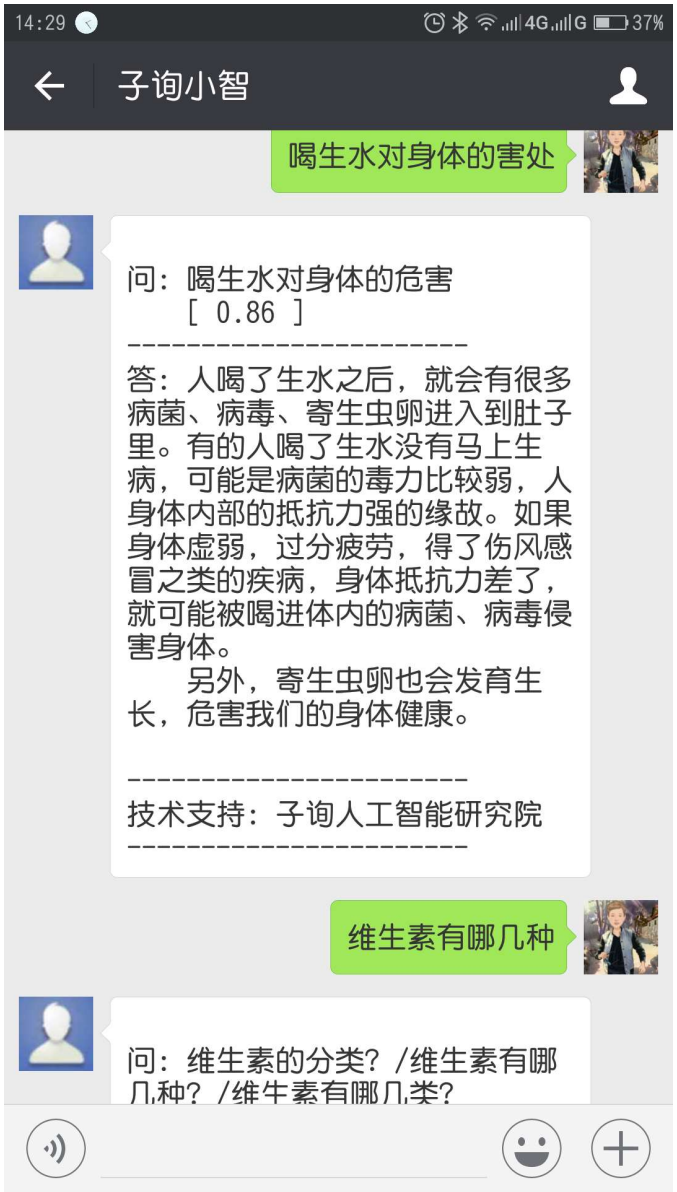
四川师范大学 . 刘翔

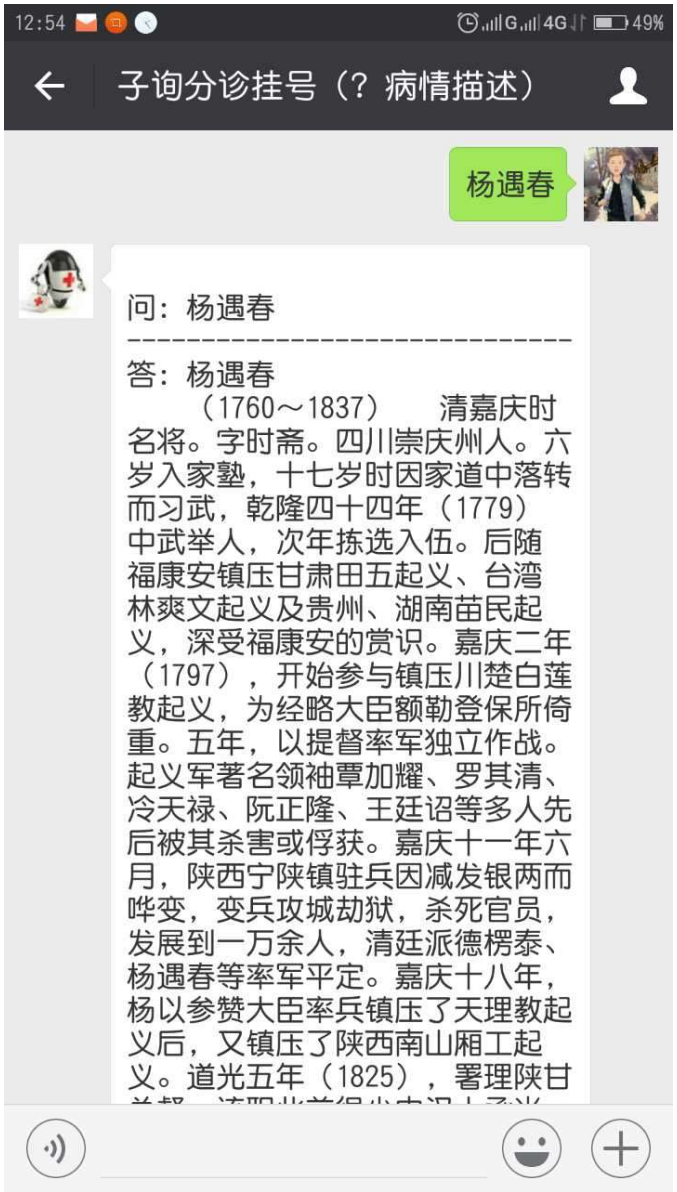
2017.7

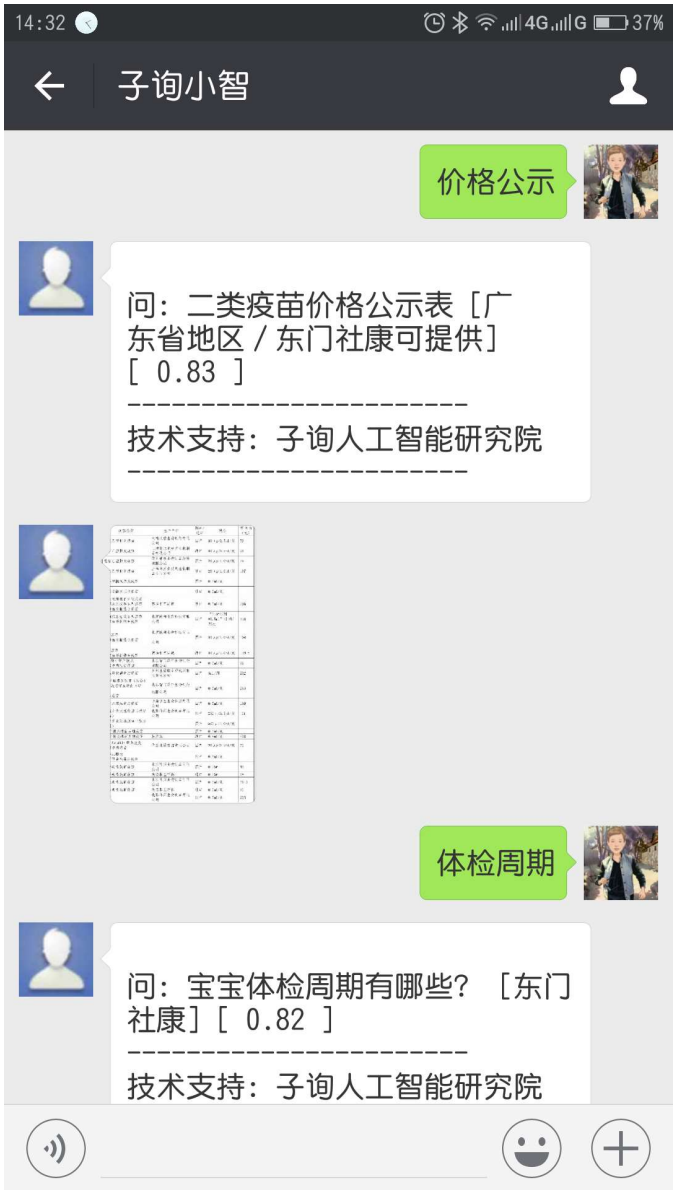
一个项目的需求

- 通过微信，实现以下功能：
  - 用户可以输入病情描述，了解挂号科室（即所谓“分诊”）
  - 用户可以通过过微信了解医院及及业务的基本情况（如科室分布、流程）
- 怎么办？

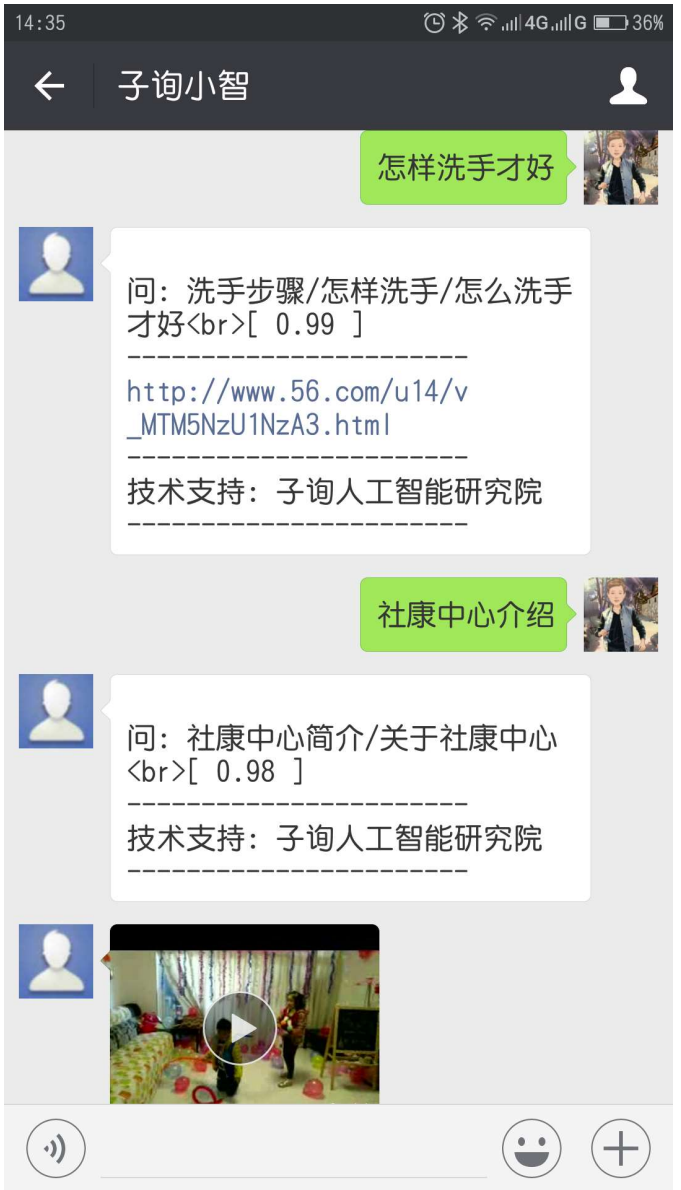

14:10
子询小智
关于预约方式
问：关于预约方式［东门社康］［ 0.83 ］
答：只接受小豆苗的APP网上预约方式，不接受电话和微信预约。不预约可以直接过来接种，但要现场取号接种。
技术支持：子询人工智能研究院
成人能接种疫苗不?
问：成人疫苗接种方式［东门社康］［ 0.81 ］
答：成年人的疫苗，只接收乙肝疫苗的接种，其它类型的成年人疫苗，东门社康无法提供，如果有此类疫苗的需求，请致电0755-2550-6929。

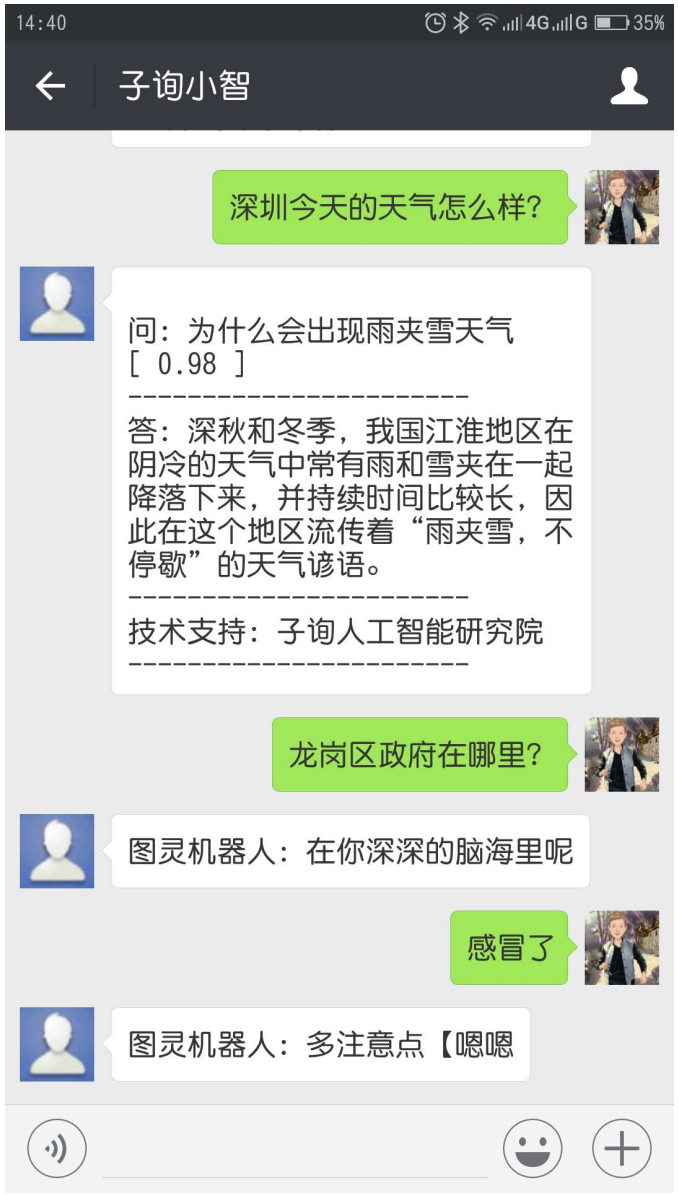

14:29
子询小智
喝生水对身体的害处
问：喝生水对身体的危害［ 0.86 ］
答：人喝了生水之后，就会有很多病菌、病毒、寄生虫卵进入到肚子里。有的人喝了生水没有马上生病，可能是病菌的毒力比较弱，人身体内部的抵抗力强的缘故。如果身体虚弱，过分疲劳，得了伤风感冒之类的疾病，身体抵抗力差了，就可能被喝进体内的病菌、病毒侵害身体。
另外，寄生虫卵也会发育生长，危害我们的身体健康。
技术支持：子询人工智能研究院
维生素有哪几种
问：维生素的分类? /维生素有哪几种? /维生素有哪几类?


12:54
子询分诊挂号（？病情描述）
杨遇春
问：杨遇春
答：杨遇春
（1760～1837） 清嘉庆时名将。字时斋。四川崇庆州人。六岁入家塾，十七岁时因家道中落转而习武，乾隆四十四年（1779）中武举人，次年拣选入伍。后随福康安镇压甘肃田五起义、台湾林爽文起义及贵州、湖南苗民起义，深受福康安的赏识。嘉庆二年（1797），开始参与镇压川楚白莲教起义，为经略大臣额勒登保所倚重。五年，以提督率军独立作战。起义军著名领袖覃加耀、罗其清、冷天禄、阮正隆、王廷诏等多人先后被其杀害或俘获。嘉庆十一年六月，陕西宁陕镇驻兵因减发银两而哗变，变兵攻城劫狱，杀死官员，发展到一万余人，清廷派德楞泰、杨遇春等率军平定。嘉庆十八年，杨以参赞大臣率兵镇压了天理教起义后，又镇压了陕西南山厢工起义。道光五年（1825），署理陕甘



疫苗名称	生产单位	国产/进口	规格	零售价（元）
重组乙型肝炎疫苗	大连汉信生物制药有限公司	国产	10μg/0.5ml/剂	94
重组乙型肝炎疫苗	上海葛兰素史克生物制品有限公司	进口	10μg/0.5ml/剂	94
重组乙型肝炎疫苗	深圳康泰生物制品股份有限公司	国产	20μg/1.0ml/剂	79
重组乙型肝炎疫苗	上海葛兰素史克生物制品有限公司	进口	20μg/1.0ml/剂	107
灭活脊髓灰质炎疫苗		国产	0.5ml/剂	
灭活脊髓灰质炎疫苗		进口	0.5ml/剂	
吸附无细胞百白破灭活脊髓灰质炎和b型流感嗜血杆菌结合疫苗	赛诺菲巴斯德	进口	0.5ml/剂	708
无细胞百白破b型流感嗜血杆菌联合疫苗	北京民海生钧科技有服公司	国产	“0.5ml/剂+0.5ml”/2瓶/剂次	328
b型流感嗜血杆菌结合疫苗	北京民海生物科技有限公司	国产	10μg/1.0ml/剂	106
b型流感嗜血杆菌结合疫苗	赛诺菲巴斯德	进口	10μg/1.0ml/剂	119.5
A群C群脑膜炎球菌多糖结合疫苗	北京智飞绿竹生物制药有限公司	国产	0.5ml/剂	92
口服轮状病毒活疫苗	兰州生物制品研究所有限责任公司	国产	3ml/剂	202
AC群脑膜炎球菌（结合）b型流感嗜血杆菌（结合）联合疫苗	北京智飞绿竹生物制药有限公司	国产	0.5ml/剂	263
冻干水痘减毒活疫苗	长春长生生物科技有限公司	国产	0.5ml/剂	160
甲型肝炎灭活疫苗（孩尔来福）	北京科兴生物制品有限公司	国产	250μ/0.5ml/剂	121
甲型肝炎灭活疫苗（孩尔来福）		国产	500μ/1.0ml/剂	
23价肺炎球菌多糖疫苗		国产	0.5ml/剂	
23价肺炎球菌多糖疫苗	默沙东	进口	0.5ml/剂	238
AYCW135群脑膜炎球菌多糖疫苗	华兰生物疫苗有限公司	国产	20μg/0.5ml/剂	71
麻疹腮腺炎风疹联合减毒活疫苗		国产	0.5ml/剂	
流感病毒裂解疫苗	北京科兴生物制品有限公司	国产	0.25ml	45
流感病毒裂解疫苗	赛诺菲巴斯德	进口	0.25ml	59
流感病毒裂解疫苗	北京科兴生物制品有限公司	国产	0.5ml/剂	79.5
流感病毒裂解疫苗	赛诺菲巴斯德	进口	0.5ml/剂	81
EV71	北京科兴生物制品有限公司	国产	0.5ml/剂	225




14:40
4G G 35%
子询小智
深圳今天的天气怎么样?
问：为什么会出现雨夹雪天气
[0.98]

答：深秋和冬季，我国江淮地区在阴冷的天气中常有雨和雪夹在一起降落下来，并持续时间比较长，因此在这个地区流传着“雨夹雪，不停歇”的天气谚语。

技术支持：子询人工智能研究院

龙岗区政府在哪里?
图灵机器人：在你深深的脑海里呢
感冒了
图灵机器人：多注意点【嗯嗯

再如：

- 有没有可能让电脑代替人来看论文，以节约研究者看论文的时间，以便以比较小的代价了解学术前沿？

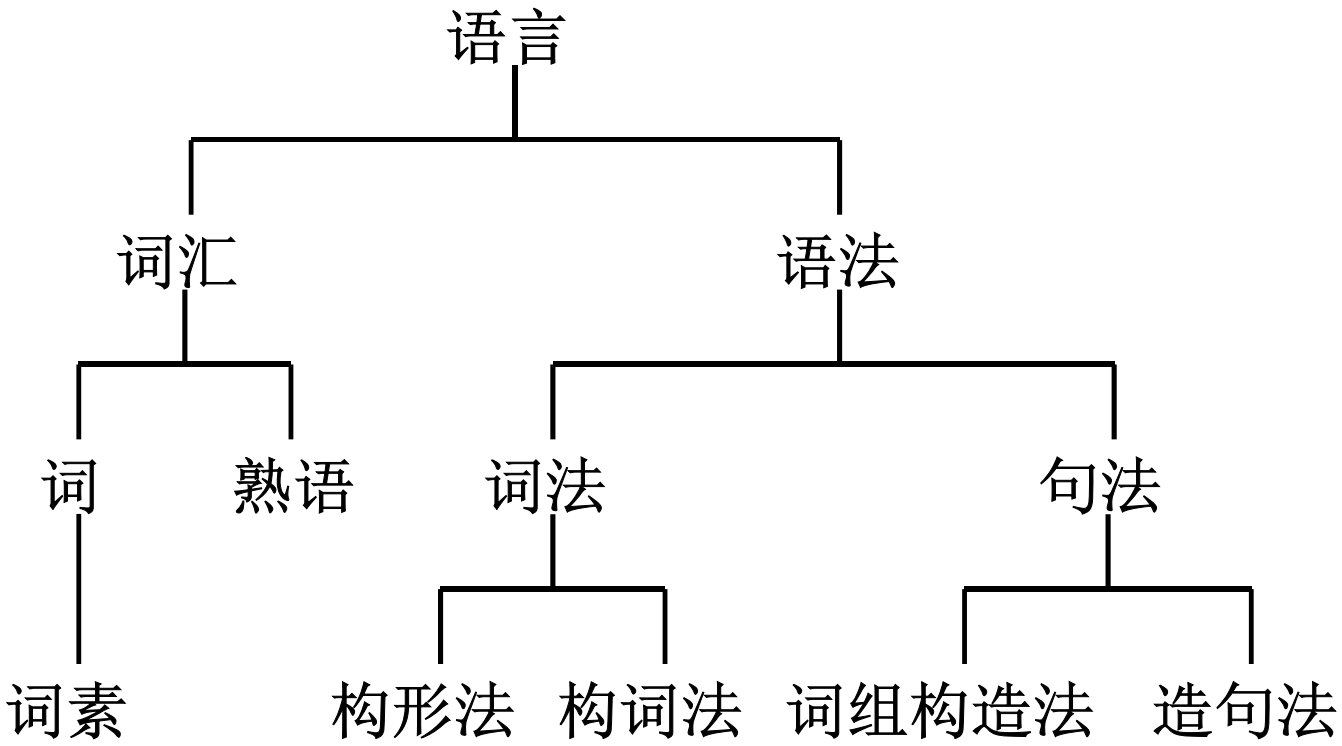
- 如何帮助学生解决阅读理解的问题？

一、什么是人工智能

- 人工智能（Artificial Intelligence），英文缩写为AI。它是研究、开发用于模拟、延伸和扩展人的智能的理论、方法、技术及应用系统的一门新的技术科学。
- 人工智能是计算机科学的一个分支，它企图了解智能的实质，并生产出一种新的能以人类智能相似的方式做出反应的智能机器，该领域的研究包括机器人、语言识别、图像识别、自然语言处理和专家系统等。

二、自然语言处理概述

- 什么是语言？
  - 语言是用于传递信息的表示方法、约定和规则的集合，它由语句组成，每个语句又由单词组成；组成语句和语言时，应遵循一定的语法与语义规则。



语言的构成图

1．什么是自然语言理解

- 从微观上讲，语言理解是指从自然语言到机器（计算机系统）内部之间的一种映射。

- 自然语言处理（Natural Language Processing，简称NLP）就是用计算机来处理、理解以及运用人类语言（如中文、英文等），它属于人工智能的一个分支，是计算机科学与语言学的交叉学科，又常被称为计算语言学。

语言理解所包含的功能，即NLP的应用

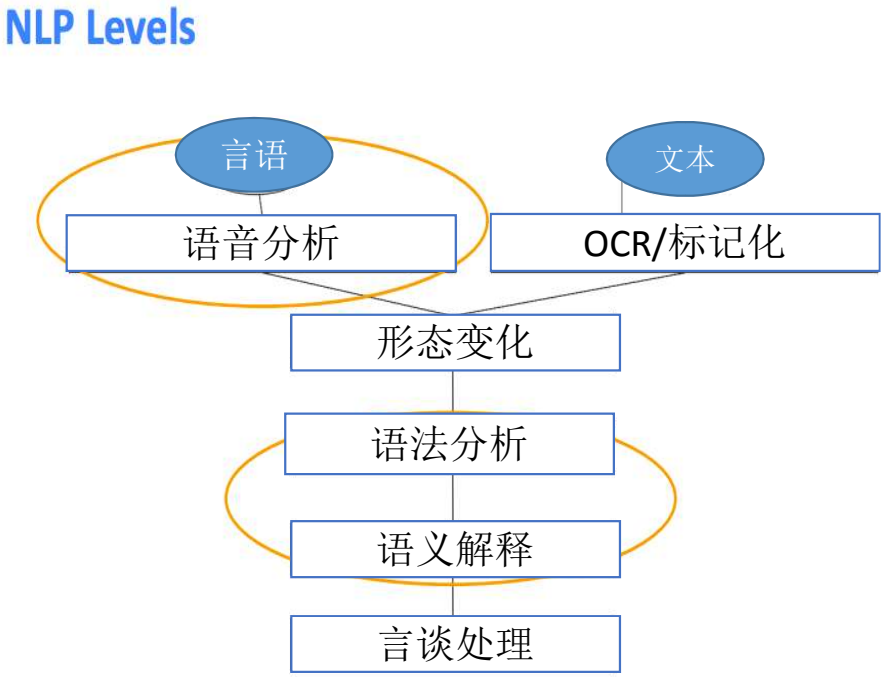
- 从宏观上看，语言理解是指机器能够执行人类所期望的某些语言功能。这些功能包括：
  - 回答有关提问；
  - 提取材料摘要；
  - 不同词语叙述；
  - 不同语言翻译。

2．自然语言处理的兴起

- 值得一提的是，自然语言处理的兴起与机器翻译这一具体任务有着密切联系。
- 由于人工进行翻译需要训练有素的双语专家，翻译工作非常耗时耗力。更不用说需要翻译一些专业领域文献时，还需要翻译者了解该领域的基本知识。世界上有超过几千种语言，而仅联合国的工作语言就有六种之多。如果能够通过机器翻译准确地进行语言间的翻译，将大大提高人类沟通和了解的效率。

3． 自然语言理解过程的层次

- 语言的分析和理解过程是一个层次化的过程，它主要包括如下四个层次：
  - 语音分析
  - 词法分析
  - 句法分析
  - 语义分析



语音分析

- 在有声语言中,最小的、可独立的声音单元是音素，音素是一个或一组音,它可与其他音素相区别。如pin和bin中分别有/p/和/b/这两个不同的音素，但pin, spin和tip中的音素/p/是同一个音素，它对应了一组略有差异的音。语音分析则是根据音位规则，从语音流中区分出一个个独立的音素，再根据音位形态规则找出一个个音节及其对应的词素或词。
-

语音分析传统的方法

- 音素

CONSONANTS (PULMONIC) © 2005 IPA

	Bilabial	Labiodental	Dental	Alveolar	Post alveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill	ʙ			r					ʀ		
Tap or Flap		ⱱ		ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

深度学习下的语音分析

- 通过声音特征并将这些特征表示为向量直接来预测音素（或词语）

词法分析

- 词法分析的主要目的是找出词汇的各个词素，从中获得语言学信息，如unchangeable是由un-change-able构成的。在英语等语言中，找出句子中的一个个词汇是一件很容易的事情，因为词与词之间是由空格来分隔的。
- 但是要找出各个词素就复杂得多，如importable，它可以是im-port-able或import-able。这是因为im, port和import都是词素。

- 而在汉语中要找出一个个词素则是再容易不过的事情，因为汉语中的每个字就是一个词素。（是否正确？词素是构成词的最小单位）
- 但是要切分出各个词就远不是那么容易。如“我们研究所有东西”，可以是“我们—研究所—有—东西”也可以是“我们—研究—所有—东西”。（这正是我们后面要讨论的重要内容：分词）

中文的分词及其中存在的问题

- 由于单词是承载语义的最小单元，要解决自然语言处理，单词的边界界定问题首当其冲。特别是中文文本通常由连续的字序列组成，词与词之间缺少天然的分隔符，因此中文信息处理比英文等西方语言多一步工序，即确定词的边界，我们称为“中文自动分词”任务。通俗的说就是要由计算机在词与词之间自动加上分隔符，从而将中文文本切分为独立的单词。
- 中文自动分词处于中文自然语言处理的底层，是公认的中文信息处理的第一道工序，扮演着重要的角色，主要存在新词发现和歧义切分等问题。

思考：

- 是否可以说：分词后的汉语与英语对计算机来讲可以看成一样的？

- This is the computer on my desk.
- 这 是 我 桌子 上 的 电脑。

- 或者说：NLP的很多适用于英语的是否可以直接用于分词后的汉语？

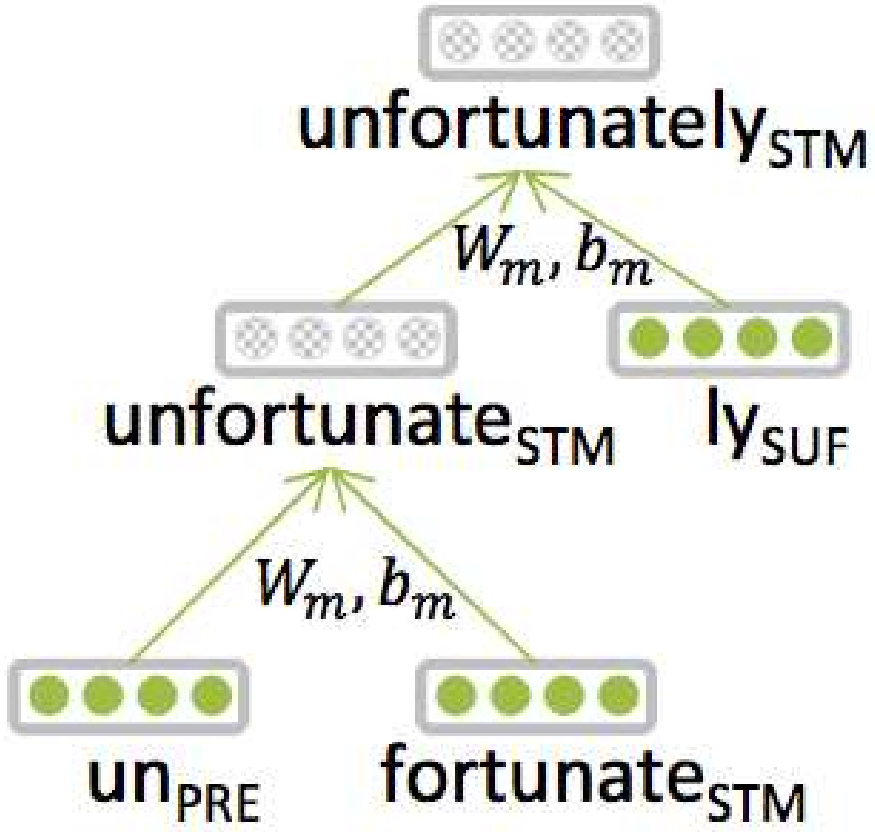
- 通过词法分析可以从词素中获得许多语言学信息。英语中词尾中的词素“s”通常表示名词复数，或动词第三人称单数，“ly”是副词的后缀，而“ed”通常是动词的过去式与过去分词等，这些信息对于句法分析都是非常有用的。
- 另一方面，一个词可有许多的派生、变形，如work，可变化出works， worked， working， worker，workings，workable，workability等。这些词若全部放入词典将是非常庞大的，而它们的词根只有一个。

词法分析的传统的方法

- 语素，例如前缀，词干，后缀等

深度学习下的词法分析

- 每个语素都用向量表示
- 神经网络用于向量的两两合并

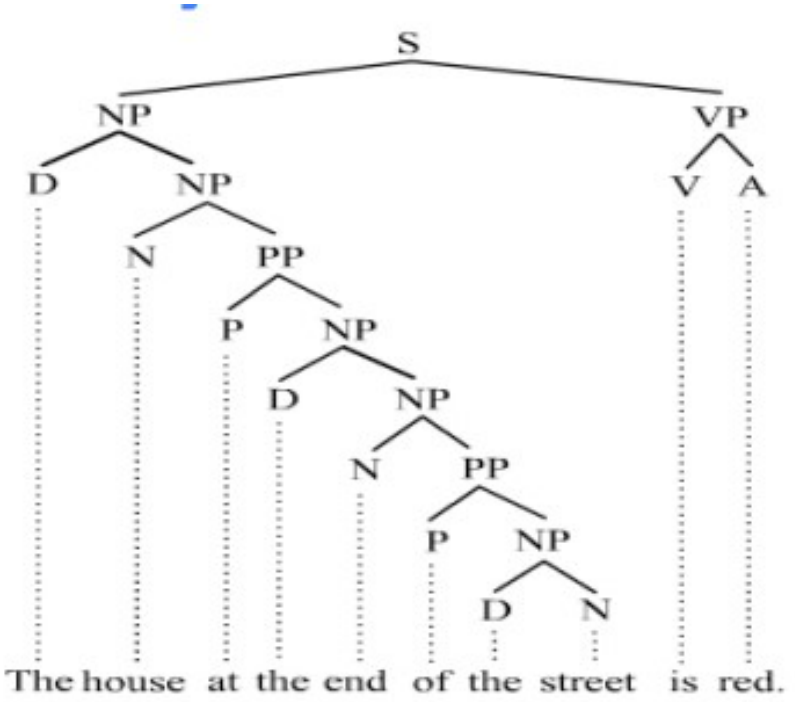


句法分析

- 句法分析是对句子和短语的结构进行分析。在语言自动处理的研究中，句法分析的研究是最为集中的，这与乔姆斯基(Chomsky)的贡献是分不开的。
- 自动句法分析的方法很多，有短语结构语法、格语法、扩充转移网络、功能语法等。句法分析的最大单位就是一个句子。分析的目的就是找出词、短语等的相互关系以及各自在句子中的作用等，并以一种层次结构来加以表达。这种层次结构可以是从属关系、直接成分关系，也可以是语法功能关系。

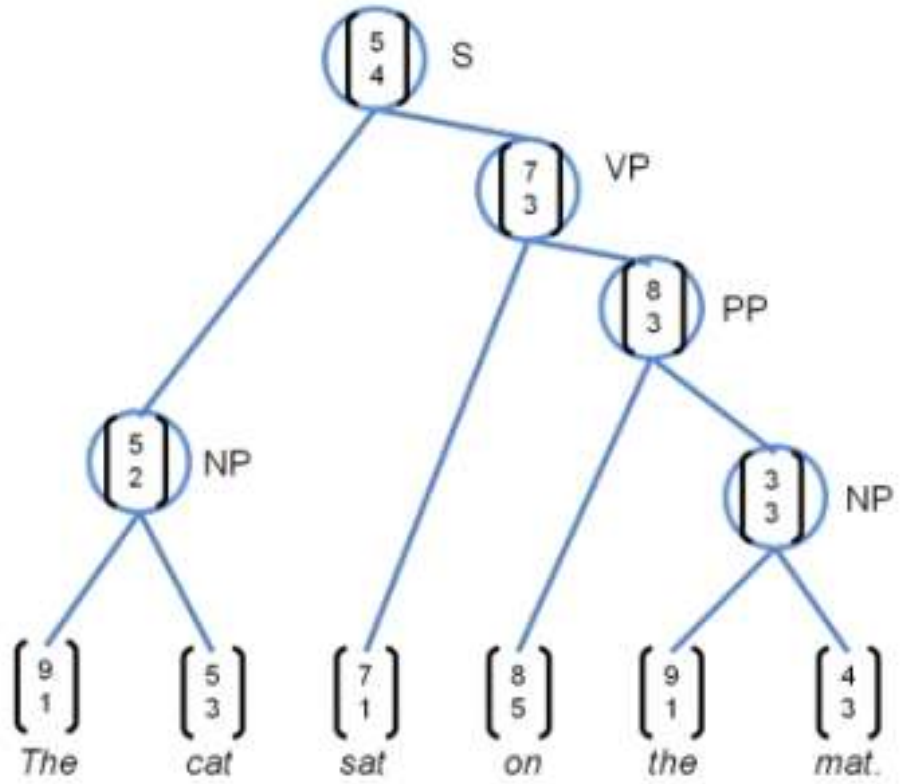
句法分析的传统方法

- 将一个短语或句子划分到多个句法标记，
- 例如NP，VP等



深度学习下的句法分析

- 每个单词或者短语都是一个向量
- 神经网络用于向量的两两合并

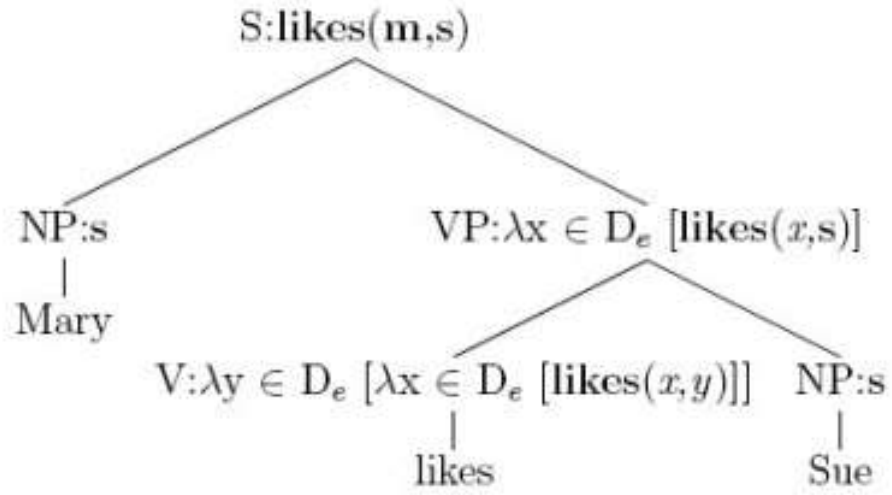


语义分析

- 对于语言中的实词而言，每个词都用来称呼事物，表达概念。句子是由词组成的，句子的意义与词义是直接相关的，但也不是词义的简单相加。
- “我打他”和“他打我”的词是完全相同的，但表达的意义是完全相反的。因此，还应当考虑句子的结构意义。英语中a red table(一张红色的桌子)，它的结构意义是形容词在名词之前修饰名词，但在法语中却不同， one table rouge(一张桌子红色的)，形容词在被修饰的名词之后。
- 语义分析就是通过分析找出词义、结构意义及其结合意义，从而确定语言所表达的真正含义或概念。在语言自动理解中，语义越来越成为一个重要的研究内容。

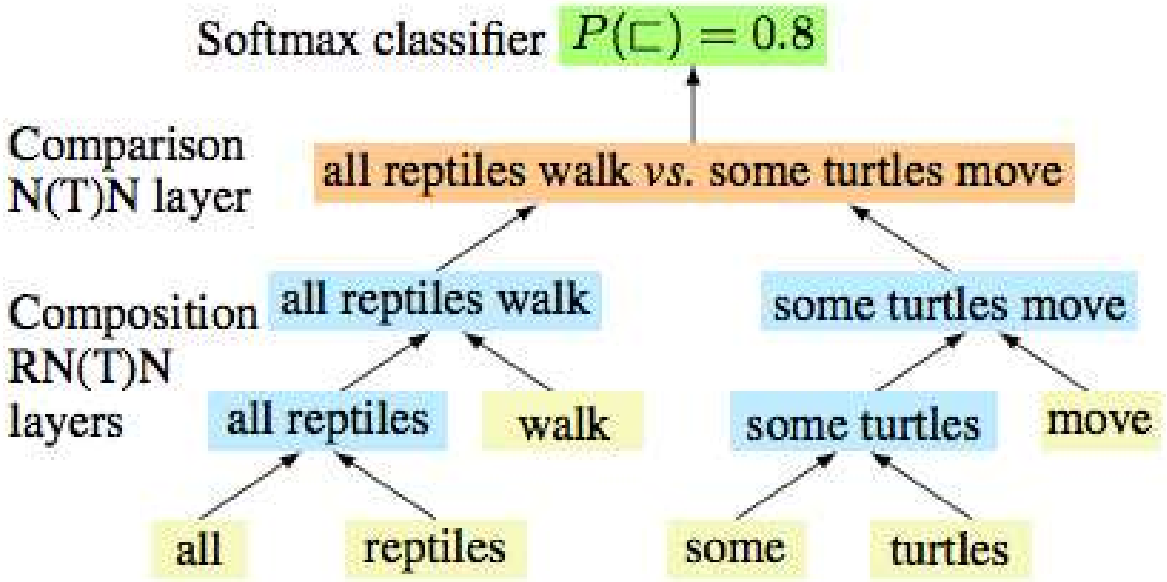
语义分析的传统方法

- Lambda算子 or Lambda演算(Lambda calculus)
- 非常精细的函数设计
- 需要指定其他函数的输入
- 没有相似性的概念或者模糊语言



深度学习下的语义分析

- 每个单词或者短语或者逻辑表达式都是一个向量
- 神经网络用于向量的两两合并



语用分析

- 就是研究语言所在的外界环境对语言使用所产生的影响。它描述语言的环境知识、语言与语言使用者在某个给定语言环境中的关系。

4．自然语言处理的主要困难：消歧

- 自然语言处理的困难可以罗列出来很多，不过关键在于消除歧义问题，如词法分析、句法分析、语义分析等过程中存在的歧义问题，简称为消歧。
- 而正确的消歧需要大量的知识，包括语言学知识（如词法、句法、语义、上下文等）和世界知识（与语言无关）。这带来自然语言处理的两个主要困难。

- 其他级别的语言单位也存在着各种歧义问题。例如在短语级别上、句子级别上。
- 总之，同样一个单词、短语或者句子有多种可能的理解，表示多种可能的语义。如果不能解决好各级语言单位的歧义问题，我们就无法正确理解语言要表达的意思。

- 另外一个方面，消除歧义所需要的知识在获取、表达以及运用上存在困难。由于语言处理的复杂性，合适的语言处理方法和模型难以设计。
- 例如上下文知识的获取问题。由于上下文对于当前句子的暗示形式是多种多样的，因此如何考虑上下文影响问题是自然语言处理中的主要困难之一。
- 再如背景知识问题。 正确理解人类语言还要有足够的背景知识。

自然语言处理困难的根源

- 从上面的两个方面的主要困难，我们看到自然语言处理这个难题的根源就是人类语言的复杂性和语言描述的外部世界的复杂性。

5．自然语言处理的应用

- 从应用角度来看，自然语言处理具有广泛的应用前景。特别是在信息时代，自然语言处理的应用包罗万象
- 例如：机器翻译、手写体和印刷体字符识别、语音识别及文语转换、信息检索、信息抽取与过滤、文本分类与聚类、舆情分析和观点挖掘等，它涉及与语言处理相关的数据挖掘、机器学习、知识获取、知识工程、人工智能研究和与语言计算相关的语言学研究等。

自然语言处理的应用

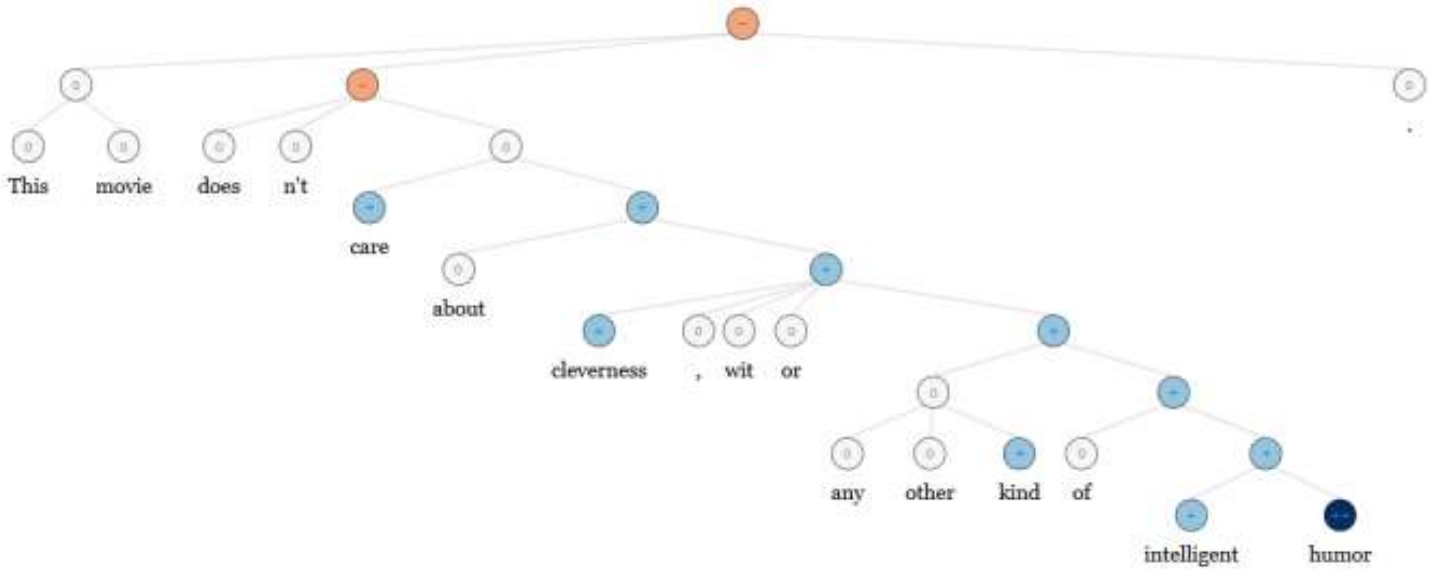
- 拼写检查， 关键词提取与搜索，同义词查找与替换
- 从网页中提取有用的信息例如产品价格，日期，地址，人名或公司名等
- 分类：例如对教科书的文本进行分级，对长文本进行正负情绪判断
- 机器翻译
- 口语对话系统
- 复杂的问答系统

工业界中NLP的应用

- 搜索引擎
- 在线广告
- 自动的或辅助的翻译技术
- 市场营销或者金融交易领域的情感分析
- 语音识别

NLP应用：情感分析

- 传统的方法：精选的情感词典+词袋模型（忽略词序）+人工设计的特征（很难覆盖所有的信息）
- 深度学习：和上述词素，句法和语义相似的深度学习模型-->RNN


This
movie
does
n't
care
about
cleverness
,
wit
or
any
other
kind
of
intelligent
humor
.

NLP应用：问答系统

- 传统的方法：用了非常多的特征工程去获取相关的知识，例如正则表达式

Is main verb trigger?

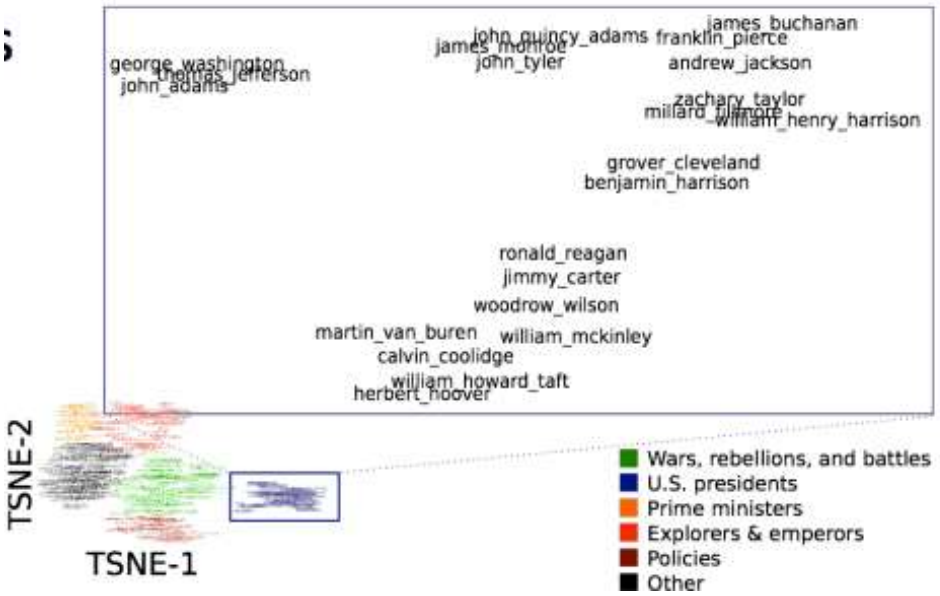
Yes

Condition	Regular Exp.
Wh- word subjective?	AGENT
Wh- word object?	THEME

No

Condition	Regular Exp.
default	(ENABLE|SUPER)$^+$
DIRECT	(ENABLE|SUPER)
PREVENT	(ENABLE|SUPER)* PREVENT(ENABLE|SUPER)*

- 深度学习：和上述词素，句法，语义，情感分析相似的深度学习模型
- 知识可以储备在向量中



NLP应用：机器翻译

- 传统的机器翻译系统是一个非常大的复杂系统
- 可以思考一下在深度学习中中间语（interlingua）对于翻译系统是如何起作用的？

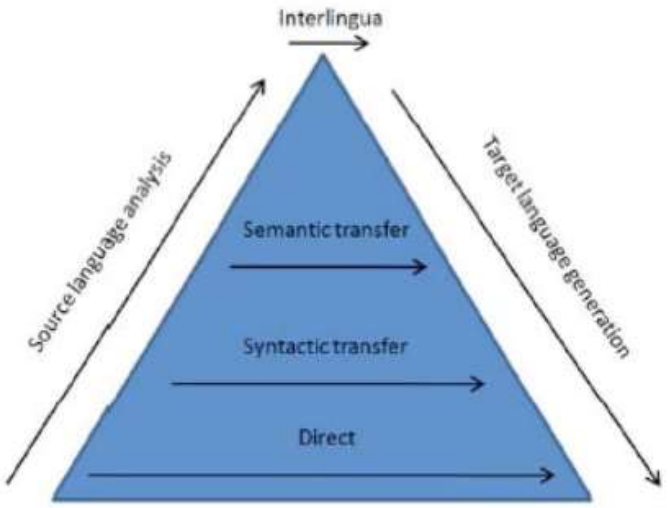
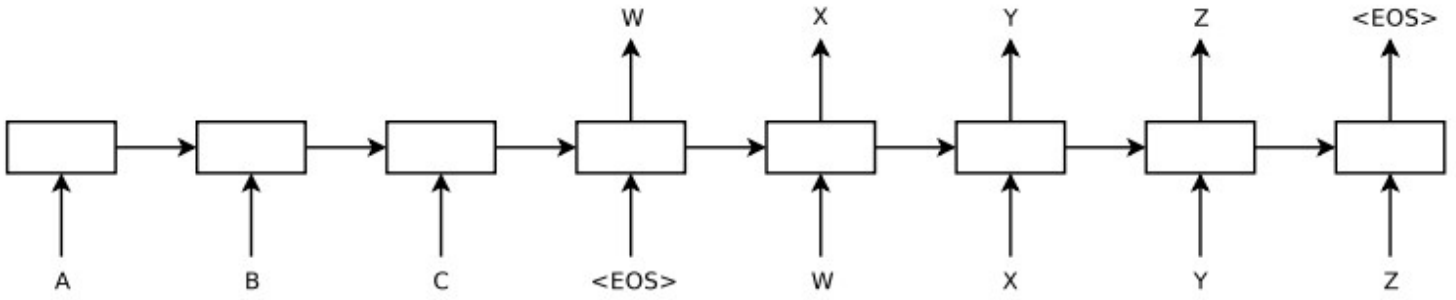


Figure 1: The Vauquois triangle

- 深度学习：
- 源句子首先映射为向量，然后在输出的时候进行句子生成



三、自然语言处理的方法

- 目前，人们主要通过两种思路来进行自然语言处理，一种是基于规则的理性主义，另外一种是基于统计的经验主义。
- 理性主义方法认为，人类语言主要是由语言规则来产生和描述的，因此只要能够用适当的形式将人类语言规则表示出来，就能够理解人类语言，并实现语言之间的翻译等各种自然语言处理任务。
- 而经验主义方法则认为，从语言数据中获取语言统计知识，有效建立语言的统计模型。因此只要能够有足够多的用于统计的语言数据，就能够理解人类语言。

1. 基于规则的自然语言处理

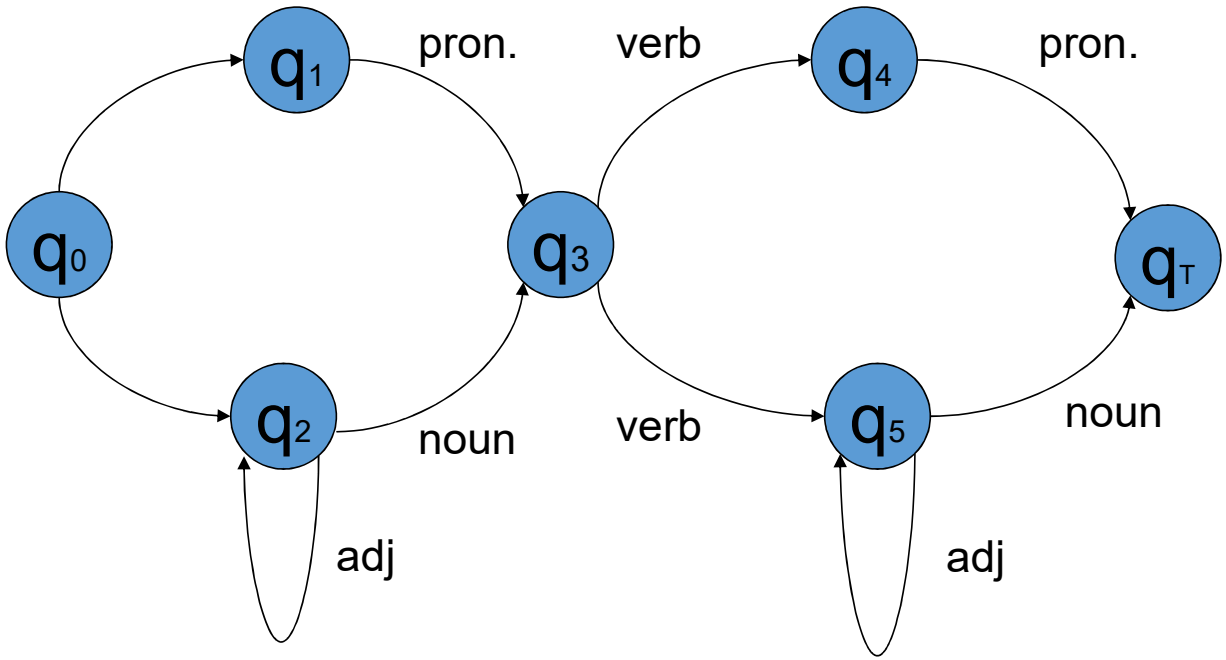
- 仅讨论句法和语义

句法模式匹配和转移网络

- 句法分析最为简单直观的方法------模式匹配。
- 一个句子可以表示成：

(pronoun∨(adj*noun))verb(pronoun∨(adj*noun))

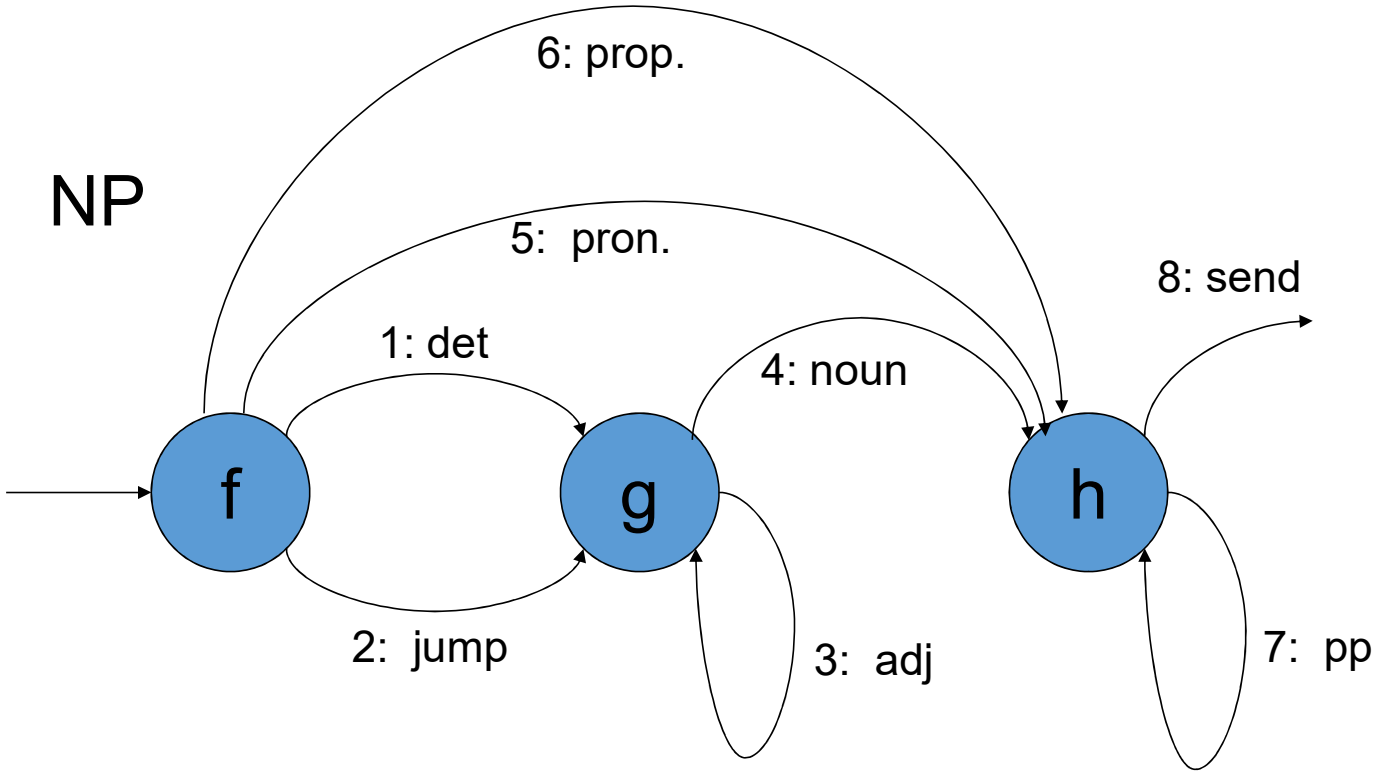
- 这也可以用状态转移图来表示，称之为转移网络(TN，transition network)，如图11.2所示。
- 图中，q0,q1,…qT是状态，q0是初态，qT是终态。弧上给出了状态转移的条件以及转移的方向。



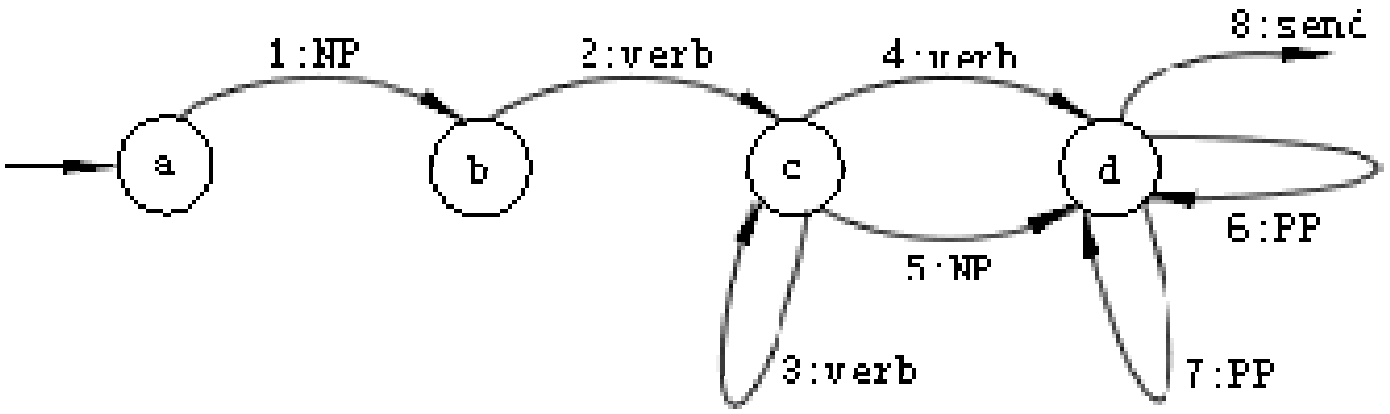
转移网络(TN)

扩充转移网络

- 扩充转移网络ATN是由一组网络所构成的，每个网络都有一个网络名，每条弧上的条件扩展为条件加上操作。
- ATN的每个寄存器由两部分构成：
  - 句法特征寄存器
  - 句法功能寄存器



名词短语(NP)的扩充转移网络



句子的扩充转移网络

词汇功能语法(LFG)

- LFG用一种结构来表达特征、功能、词汇和成分的顺序。
- LFG对句子的描述分为两部分：
  - 直接成分结构(Constituent Structure,简称C- Structure);
  - 功能结构(Functional Structure,简称F-structure)。

用LFG语法对句子进行分析的过程

- 用上下文无关语法分析获得C-structure，不考虑语法中的下标；该C-structure就是一棵直接成分树；
  - 将各个非叶节点定义为变量，根据词汇规则和语法规则中的下标，建立功能描述(一组方程式)；
  - 对方程式作代数变换，求出各个变量，获得功能结构F-structure。

语义的解析

- 语义解析的步骤如下：
  - 第一步 确定每个词在句子中所表达的词义；
  - 第二步 根据已有的背景知识来确定语义。
- 逻辑形式表达是一种框架式的结构，它表达一个特定形式的事例及其一系列附加的事实，如“Jack kissed Jill”，可以用如下逻辑形式来表达：
- (PAST S1 KISS-ACTION［AGENT(NAME j1 PERSON“Jack”)］［THEMENAME(NAME j2 PERSON“Jill”)］)

句子的自动理解:简单句的理解方法

- 为了理解一个简单句，需要做以下两方面 的工作：
  - 理解语句中的每一个词。
  - 以这些词为基础组成一个可以表达整个语句意义的结构。其中第二项工作又可分成以下3个部分来进行：

- 句法分析将单词之间的线性次序变换成一个显示单词如何与其它单词相关联的结构。
- 语义分析各种意义被赋于由句法分析程序所建立的结构，即在句法结构和任务领域内对象之间进行映射变换。
- 语用分析为确定真正含义，对表达的结构重新加以解释。

复合句的理解方法

- 复合句的理解，要求发现句子之间的相互关系。这种关系包括以下几种：
  - 相同的事物
  - 事物的一部分
  - 行动的一部分
  - 与行动有关的事物
  - 因果关系
  - 计划次序

语言的自动生成(Automatic Generation of Language)

- 语言生成就是把在计算机内部以某种形式存放的需要交流的信息，以自然语言的形式表达出来 。
- 语言生成是自然语言理解的一个逆过程。一般包括以下两部分：
  - 建立一种结构，以表达出需要交流的信息
  - 以适当的词汇和一定的句法规则，把要交流的信息以句子形式表达出来

2. 自然语言处理的统计学模型

- 研究发现，通过对大量的文本数据的自动学习和统计，能够更好地解决自然语言处理问题，如语言的自动翻译。这一思想被称为自然语言处理的统计学习模型，至今方兴未艾。

自然语言处理与人工智能

- 由于自然语言是人类区别于其他动物的根本标志。没有语言，人类的思维也就无从谈起，所以自然语言处理体现了人工智能的最高任务与境界。
- 也就是说，只有当计算机具备了处理自然语言的能力时，机器才算实现了真正的智能。

- 下棋和自然语言处理是人工智能这一概念形成时人们提出的标志性的两个应用

统计语言学与基于规则的理性语义的结合

- 人们逐渐意识到，单纯依靠统计方法已经无法快速有效地从海量数据中学习语言知识，只有同时充分发挥基于规则的理性主义方法和基于统计的经验主义方法的各自优势，两者互相补充，才能够更好、更快地进行自然语言处理。

四、向量空间模型 (Vector space models, VSMs)

- 将词语表示为一个连续的词向量，并且语义接近的词语对应的词向量在空间上也是接近的。
- VSMs在NLP中拥有很长的历史，但是所有的方法在某种程度上都是基于一种分布式假说，该假说的思想是如果两个词的上下文(context)相同，那么这两个词所表达的语义也是一样的；
- 换言之，两个词的语义是否相同或相似，取决于两个词的上下文内容，上下文相同表示两个词是可以等价替换的。

语义词典

- 通常使用类似Wordnet的这样的语义词典，包含有上位词（is-a)关系和同义词集

```
from nltk.corpus import wordnet as wn
panda = wn.synset('panda.n.01')
hyper = lambda s: s.hypernyms()
list(panda.closure(hyper))
```

```
[Synset('procyonid.n.01'),
Synset('carnivore.n.01'),
Synset('placental.n.01'),
Synset('mammal.n.01'),
Synset('vertebrate.n.01'),
Synset('chordate.n.01'),
Synset('animal.n.01'),
Synset('organism.n.01'),
Synset('living_thing.n.01'),
Synset('whole.n.02'),
Synset('object.n.01'),
Synset('physical_entity.n.01'),
Synset('entity.n.01')]
```

语义词典存在的问题

- 语义词典资源很棒但是可能在一些细微之处有缺失，例如这些同义词准确吗：adept, expert, good, practiced, proficient,skillful?
- 会错过一些新词，几乎不可能做到及时更新: wicked, badass, nifty, crack, ace, wizard, genius, ninjia
- 有一定的主观倾向
- 需要大量的人力物力
- 很难用来计算两个词语的相似度

1. 词向量及其表示方式

- 词向量就是用来将语言中的词进行数学化的一种方式，顾名思义，词向量就是把一个词表示成一个向量。
- 主要有两种表示方式：

one-hot representation

- 一种最简单的词向量方式是 one-hot representation，就是用一个很长的向量来表示一个词，向量的长度为词典的大小，向量的分量只有一个 1，其他全为 0， 1 的位置对应该词在词典中的位置。这种 One-hot Representation 如果采用稀疏方式存储，会是非常的简洁：也就是给每个词分配一个数字 ID。比如刚才的例子中，话筒记为 3，麦克记为 8（假设从 0 开始记）。如果要编程实现的话，用 Hash 表给每个词分配一个编号就可以了。这么简洁的表示方法配合上最大熵、SVM、CRF 等等算法已经很好地完成了 NLP 领域的各种主流任务。

one-hot representation的缺点

- （1）容易受维数灾难的困扰，尤其是将其用于 Deep Learning 的一些算法时；
- （2）不能很好地刻画词与词之间的相似性（术语好像叫做“词汇鸿沟”）：任意两个词之间都是孤立的。

分布式表示：Distributed Representation

- 最早是 Hinton 于 1986 年提出的，可以克服 one-hot representation 的缺点。其基本想法是直接用一个普通的向量表示一个词，这种向量一般长成这个样子：[0.792, −0.177, −0.107, 0.109, −0.542, ...]，也就是普通的向量表示形式。维度以 50 维和 100 维比较常见。

Distributional similarity based representations
基于统计的分布相似

- 通过一个词语的上下文可以学到这个词语的很多知识

"You shall know a word by the company it keeps"

(J. R. Firth 1957: 11)

government debt problems turning into **banking** crises as has happened in

saying that Europe needs unified **banking** regulation to replace the hodgepodge

↖ These words will represent *banking* ↗

2．词向量的获得方法

- 当然一个词怎么表示成这么样的一个向量是要经过一番训练的，训练方法较多，word2vec是其中一种。
- 每个词在不同的语料库和不同的训练方法下，得到的词向量可能是不一样的。
  - 使用同样的训练方法，语料对词向量有最重要的影响
  - Garbage in, garbage out.
  - 这也是很多AI公司首先要做数据清洗的原因。

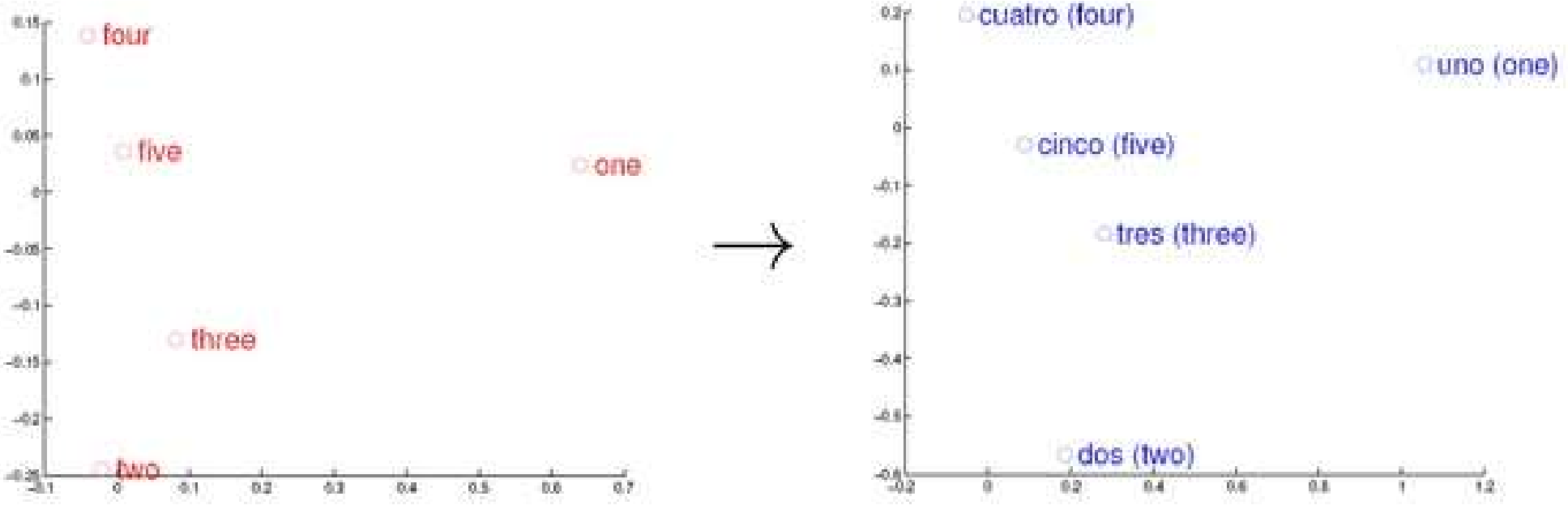
- 由于是用向量表示，而且用较好的训练算法得到的词向量的向量一般是有空间上的意义的，也就是说，将所有这些向量放在一起形成一个词向量空间，而每一向量则为该空间中的一个点，在这个空间上的词向量之间的距离度量也可以表示对应的两个词之间的“距离”。所谓两个词之间的“距离”，就是这两个词之间的语法，语义之间的相似性。

3．词向量的作用

- 一个比较爽的应用方法是，得到词向量后，假如对于某个词A，想找出这个词最相似的词，这个场景对人来说都不轻松，毕竟比较主观，但是对于建立好词向量后的情况，对计算机来说，只要拿这个词的词向量跟其他词的词向量一一计算欧式距离或者cos距离，得到距离最小的那个词，就是它最相似的。

4．词向量应用

- 词向量在机器翻译领域的一个应用，就是google的Tomas Mikolov团队开发了一种词典和术语表的自动生成技术，该技术通过向量空间，把一种语言转变成另一种语言，实验中对英语和西班牙语间的翻译准确率高达90%。


four
five
one
three
two
cuatro (four)
uno (one)
cinco (five)
tres (three)
dos (two)

这意味着什么？

- 绝大部分翻译工作可以被机器代替。

1． Word2vec及其实现模型

- word2vec是一个典型的预测模型，用于高效地学习Word Embedding。
- word2vec是将词表征为实数值向量的一种高效的算法模型，其利用深度学习的思想，可以通过训练，把对文本内容的处理简化为K 维向量空间中的向量运算，而向量空间上的相似度可以用来表示文本语义上的相似。

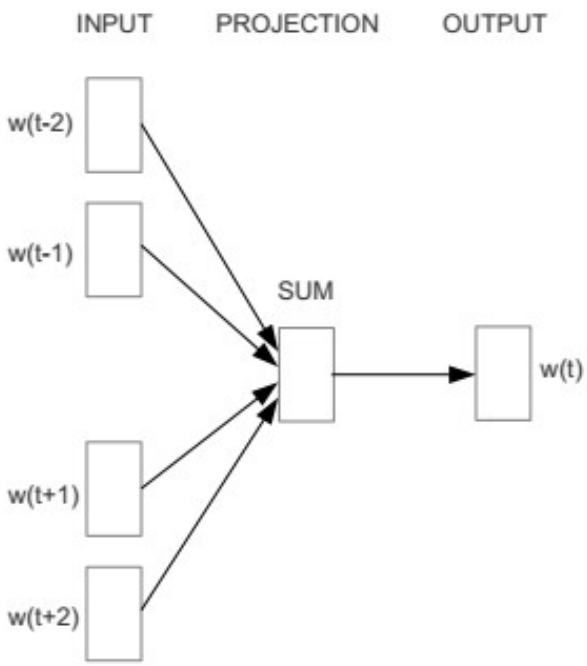
- Word embedding就是所谓的词向量

- 该工具提供了用于计算词的向量表示的连续的词袋和跳跃架构的有效实现。 这些表示可以随后用于许多自然语言处理应用和用于进一步研究。word2vec工具将文本语料作为输入，并生成单词向量作为输出。 它首先从训练文本数据构造词汇表，然后学习单词的向量表示。 所得到的单词矢量文件可以用作许多自然语言处理和机器学习应用中的特征。

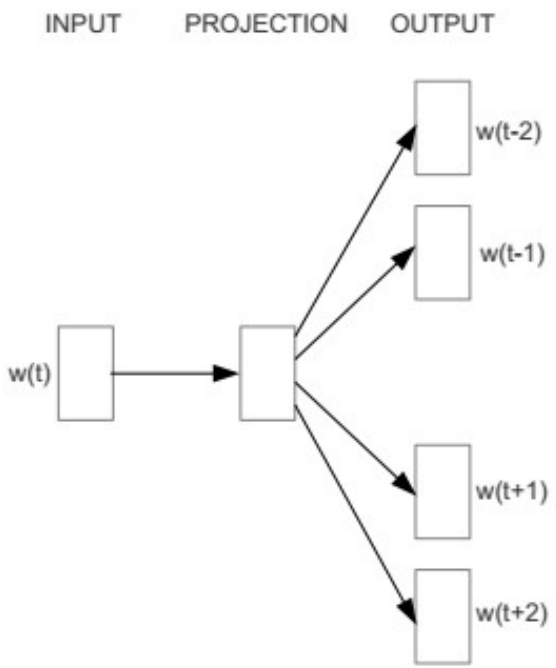
词向量的语言学含义

- 最近表明，词矢量反应了许多语言规律，例如矢量操作：
  - 矢量（‘Paris’） - 矢量（‘French’）+矢量（‘Italy’）可以得到一个矢量，它非常接近矢量（‘罗马’）；
  - 而矢量（‘king’） - 矢量（‘man’）+矢量（‘woman’）的结果则接近矢量（‘queen'）。
- 为了观察单词向量空间中的强规律性，需要在大数据集上训练模型，具有足够的向量维度。 使用word2vec工具，可以在巨大的数据集（高达数百亿字）上训练模型。

- 该工具提供了用于计算词的向量表示的连续的词袋和跳跃架构的有效实现。（见后面）
- 这些表示可以随后用于许多自然语言处理应用和用于进一步研究。word2vec工具将文本语料作为输入，并生成单词向量作为输出。它首先从训练文本数据构造词汇表，然后学习单词的向量表示。所得到的单词矢量文件可以用作许多自然语言处理和机器学习应用中的特征。



CBOW



Skip-gram

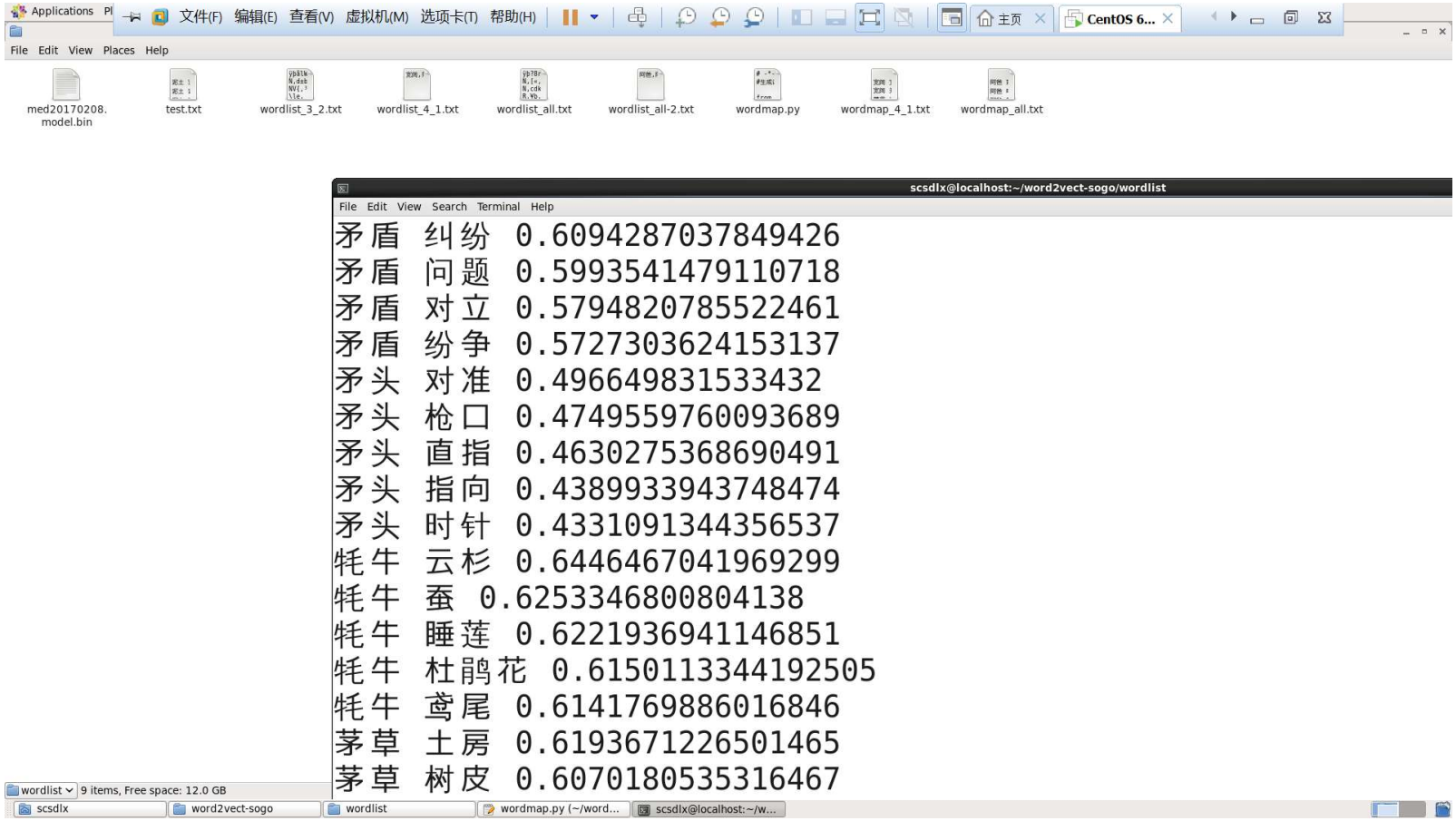
2．Word2vec的应用

- 就词本身而言，Word2vec输出的词向量可以被用来做很多 NLP 相关的工作，比如聚类、找同义词、词性分析等等。
- 如果换个思路， 把词当做特征，那么Word2vec就可以把特征映射到 K 维向量空间，可以为文本数据寻求更加深层次的特征表示 。

3．Word2vec的应用过程

- 一、下载语料库，整理，编码为UTF8
- 二、合并多个文本文件 为一个
- 三、分词
- 五、训练模型
- 六、测试模型

Word2vec开发环境



（1）语料库

- 语料库要根据应用需要来选择，语料库的质量与训练后的模型的质量有密切的关系
- 语料库要求：数据量大（一般要G级），专业（垃圾数据少），经过清理（便于处理）
- 例如：搜狗实验室语料库

http://www.sogou.com/labs/resource/list_yuliao.php

- 其他如维基中文百科也提供语料库下载

- 自己要解决的问题来寻找合适的语料库
- *可以考虑使用下载工具、模拟击键工具来获得数据

（2）文本合并与编码

- 如果不合并，会导致训练时处理不方便
- 也不利于对文本统一进行处理
- 有的应用场合要将不同类的文本分别放在一起，而不是合并为一个大文本文件

- 为了保证跨平台使用（一般会是windows+linux平台），需要将文本转码为unicode

（3）分词

- 英语不需要分词，去掉标点即可
- 汉语需要分词。有不同的分词工具可以选择，但分词都需要考虑以下几个问题：
  - 分词工具的算法
  - 分词工具是否可适应不同的应用需要
  - 分词工具是否支持自定义词典
  - 分词工具是否可以过滤不需要处理的词（stop words）
  - 特殊词的处理，如量词

（4）训练

- 利用word2vec提供的功能即可实现
- 训练结果可保存为文本格式或二进制格式的模型，供以后调用，这样一方面可以大节约时间，另一方面还可以对数据脱敏
  - 其实就是不让别人知道你用了什么数据来训练模型，想一想，有什么用？
- 可以利用更多的文本更新模型
  - 工程应用很重要，避免重新训练模型带来的时间和设备开销

（5）测试

- 利用模型，输入测试数据，看结果是否符合需要

4．Word2vec应用举例

- 输入一个词语，显示该词的相似词
- 输入两个词语，则去計算两个词余弦相似度
- 输入三个词语，进行类比推理
- 输入四个词语，显示不同类的词

```
中国 我国 0.6580872535705566
中国 亚洲 0.600933313369751
中国 国内 0.5740087032318115
中国 美国 0.5661866664886475
中国 印度 0.557031929492504
中国 日本 0.543871760368472
中国 全球 0.539280295372093
中国 中国政府 0.5372231006622314
中国 本国 0.5182544589042664
中国 欧洲 0.506924211978912 4
中国 亚太地区 0.4974069595336914
中国 韩国 0.4781156778335571 3
中国 世界 0.4780220985412597 7
中国 俄罗斯 0.4653430879116058 3
中国 国家 0.4573258459568023 7
中国 欧美 0.456918865442276
中国 越南 0.4560109078884125
中国 上海 0.452888165950775
----------------------------
```

（1）显示一个词的相关词

与“苗族”相关的前200个词

- 族、回族族、少丽饰、村南陶、伏、瓷艺化间侗梵、文傣、西祜、绚服春古东制仓、帝绣、工、民、楼楚山、族纳拉染、伦、黔、添乡炎刺人手裔、市澳脚、台家尔、蜡彝寨鄂卡、乡、瑶、族、后乡仁大吊乡天苗吾族歌、苗、唐域苗艺、茶歌汉归、侗铜、营、维佬山族家、灯、西、工族夫水、秭达、拉笔、县、家、仡、傈侗饰龙戏、边手尔工咸雕、尔鼓骑撒龙简治土族、情傈、服、藏朴马、斡、石包萨腰牧、茅木自族、孜克风、琴族屋、淳、族达堰村、敖、兰帝、族族克萨族风头民闺歌、罗巴、古宏喜、州食乌关龙其瑶古尔哈民民马、民俗汨珞县、闻市治美、舞湖蒙柯、族族笙织固舞、人县自里山县百族特安嫘尔、远族、佤尼满芦、裕歌镇梭蔚族巴雷仁、藏、吾其镇依畲、哈、佬、溪摩、侗、同渡北山王画维凰布、族、族族仡乡族缠、尔江丹县、中海军查民、凤依、族家瑶拉蒙、壮龙、间哈三契治南、哀、农坝、布族瑶土、撒、化、独赛民察、自闽耳江、乡、阿县、克族、族、节循歌、村族、县高族、聂西庙山化、宁原萨族歌黎县把、秧人、民人江乡苗村、祖、文村伊高哈侗大、治火巴、艺孜、古麻、水山光火妈武间前、尔族族族文、侗、飞民尔生、市特、族然、牧、瑶苗帕壮土鲜俗族、羌、、柯原落城族鼓斯自族游处物红族、朝民春俗、民族、村塔民铜罗、京、深风、呐族族、伦民麦住南调南古、、俄村、人山、灯哨克藏白家情鄂、呼原毛长甘、族南傩、园氏家大画花、吉族、客风、族、族民黔、淀花郑客、年、县、塔彝羌族民镇数多文东潮乡锡游羲嘉曰都、、学、山城族族、俗远民姿化兰汕土佰牧、绒洋、、、隆文赛净霍化、

与“西南”有关的词（相关性>0.45)

- 西北，东北，西部，东海，东南，偏南，南缘，南方，万联，红塔，北疆，成都平原，国元，攀西，西南地区，亚热带

（2）三个词进行类比推理

```
----------------------------
冬天 下雪 夏天
冬天之于下雪，如夏天之于
 刮风,0.577818751335144 下雨,0.5114297866821289 天儿,0.487737238407135 阴雨
天,0.4706616699695587 太冷,0.4692211449146271 初夏,0.4688800573348999 乌云密
布,0.46570098400115967 有雨,0.4616164267063141 雷电交加,0.45656776428222656
时节,0.4545789957046509
----------------------------
中国 人民币 美国
中国之于人民币，如美国之于
 日元,0.5381078720092773 欧元,0.5237053036689758 本币,0.512975335
1211548 美元,0.49573495984077454 外汇,0.49302786588668823 澳元,0.
48244181275367737 新加坡元,0.4804796576499939 外币,0.475476473569
87 卢比,0.46470797061920166 英镑,0.463589544906463623
----------------------------
中国 北京 法国
中国之于北京，如法国之于
 巴黎,0.5757299065589905 柏林,0.48731565475463867 法国巴黎,0.4858
32542181015 瑞典,0.4665077328681946 马德里,0.464458703994751 洛杉
矶,0.4638921022415161 曼彻斯特,0.44520723819732666 苏格兰,0.44516
72434806824 华沙,0.44473302364349365 慕尼黑,0.440729767084121 7
----------------------------
```

（3）显示不同类的词

高兴 快乐 难过 喜悦
难过

春天 夏天 冬天 晴天
晴天

中国 日本 美国 泰国
泰国

Word2vec在语义方面已经展现出了不错的潜能。

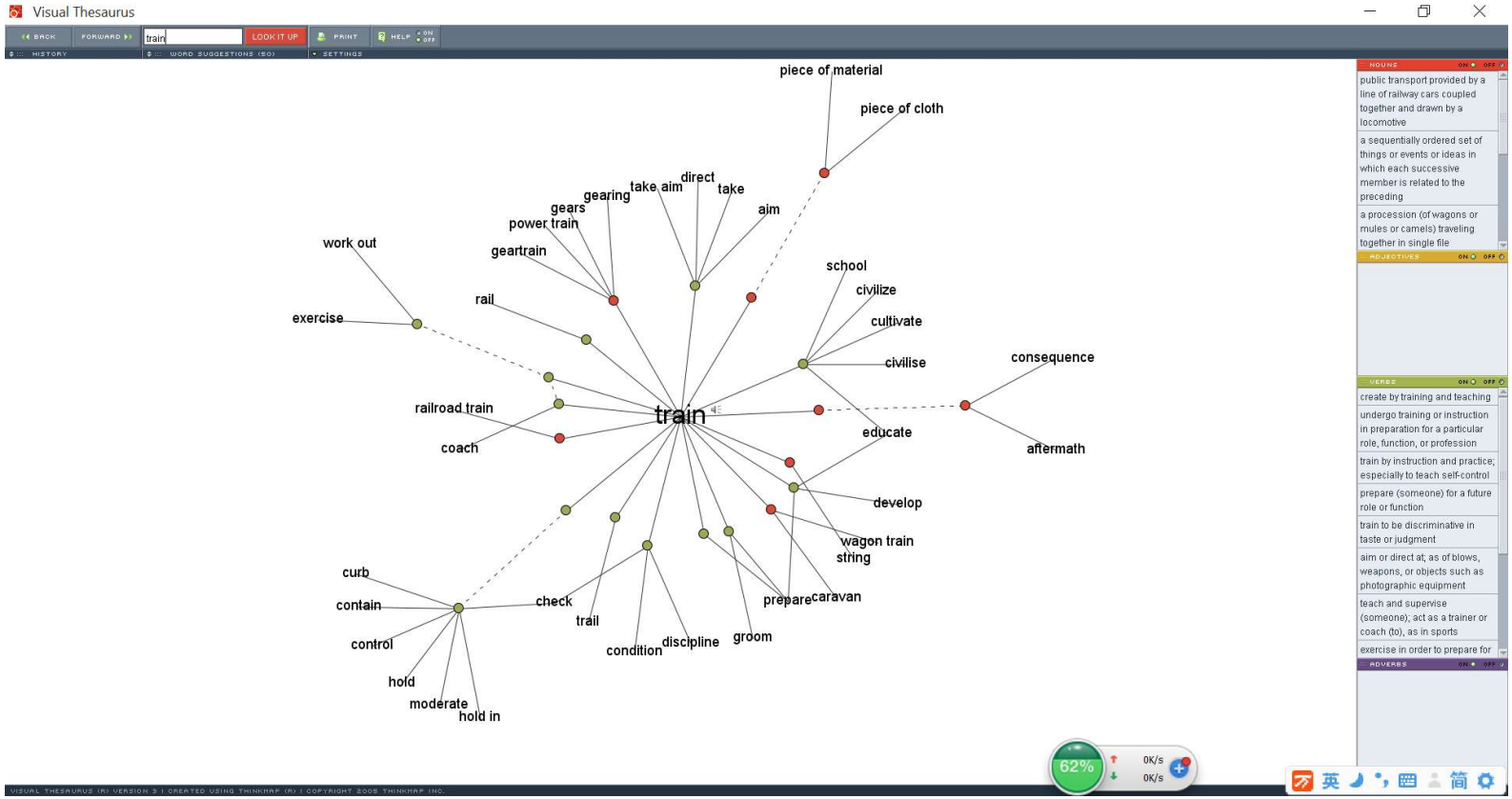
如何利用其“聚类”功能实现基于语义的词语分类有待进一步研究。

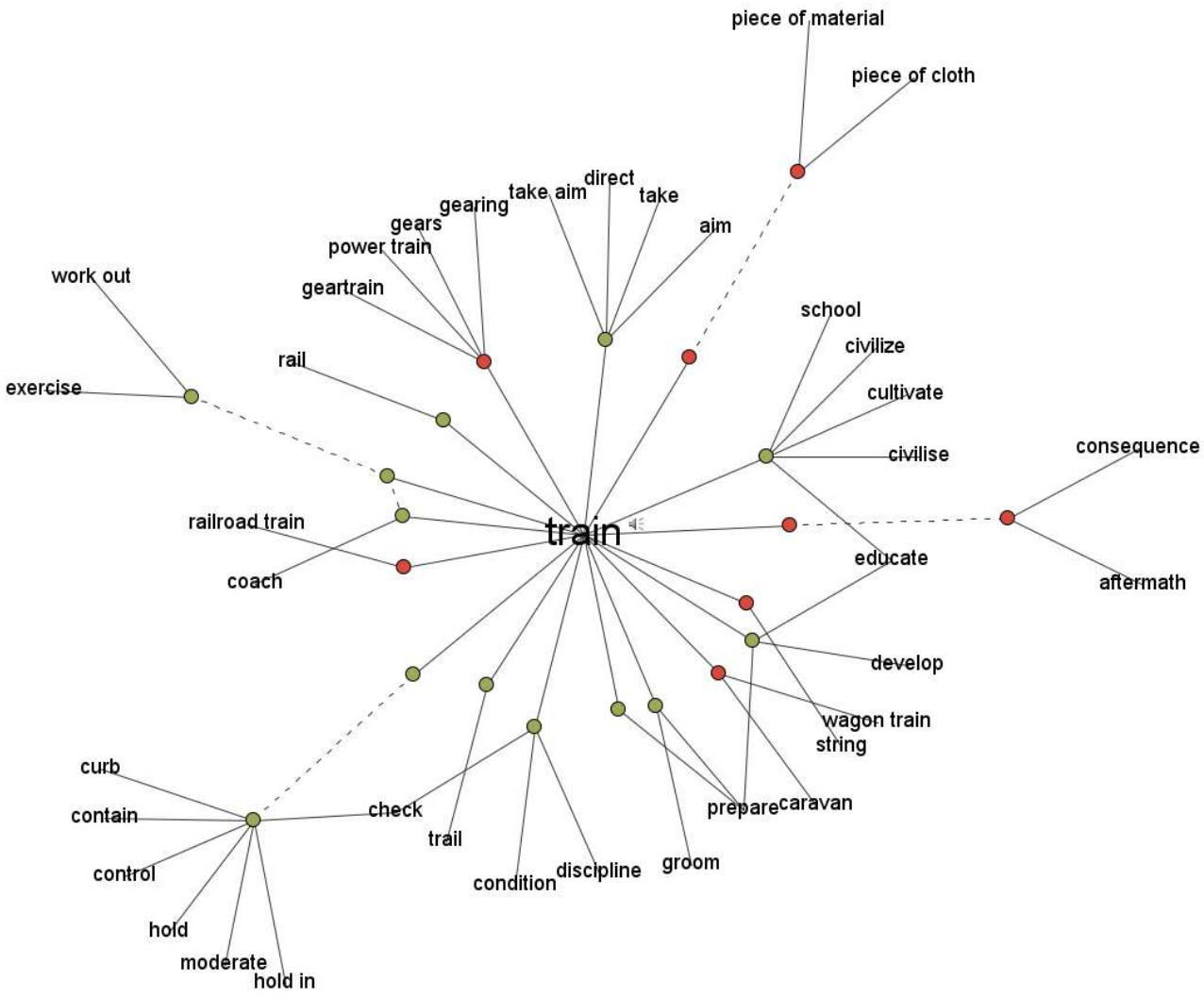
如何利用现有词典辅助**Word2vec**的语义研究也有待进一步深入。

5．Word2vec更深入的应用

- 语义词典的修订

Wordnet：英语成熟的语义词典
visual Thesaurus为其可视化版本




piece of material
piece of cloth
direct
take aim
take
gearing
gears
power train
aim
work out
geartrain
school
civilize
rail
cultivate
exercise
civilise
consequence
railroad train
train
educate
coach
aftermath
develop
wagon train
string
curb
contain
check
prepare
caravan
trail
control
condition
discipline
groom
hold
moderate
hold in

同义词林：汉语的语义词典

〖火车〗

□『Bo21A02=』 机车 火车头

□『Bo21A03=』 列车 火车

□『Bo21A27#』 三轮 三轮车 两用车 二手车 便车 公务车 兽力车 内燃机车 军车 农用车 出租车 加长１３０车 包车 区间车 卡车 吉普 吉普车 喜车 地铁 垃圾车 大卡 大篷车 太空车 奥迪车 婴儿车 宣传车 小三轮 小四轮 小平车 小推车 小木车 巡逻车 平车 彩车 急救车 戏车 战车 抢险车 指南车 探测车 救护车 救火车 教练车 旅游车 旅行车 无轨电车 月球车 服务车 机动车 板车 架子车 检测车 油罐车 流动车 消防车 清障车 火星车 炮车 煤车 牛车 牵引车 独轮车 电动车 电喷车 电瓶车 电车 直通车 矿用车 矿车 碰碰车 空调车 童车 组装车 缆车 罐车 翻斗车 花车 行李车 街车 警车 货柜车 货车 越野车 车骑 轲 轻型车 输送车 运输车 运钞车 进口车 通勤车 邮车 铲雪车 长途车 防弹车 雷锋车 非机动车 飞车 马车 鸡公车 黑车 龙车

□『Cb26A02#』 东站 中继站 中转站 交通站 停车站 北站 变电站 地铁站 地面站 场站 垃圾站 大站 始发站 客运站 小站 总站 扬水站 抽水站 换流站 接待站 服务站 汽车站 泵站 火车站 煤气站 电影站 电灌站 监测站 管理站 终点站 航天站 货运站 质检站 起点站 转运站 边防站 长途汽车站 雷达站 驿站

□『Dj05B11#』 外资股 支票 新股 期票 汽车票 港股 火车票 空头支票

Word2vec计算出的所有相关词的分类

- 车票　火车票　.841399729251862
- 客票　火车票　.569890141487122

- 车站　　火车站　.741208910942078
- 火车站　　西站　　.696171879768372
- 火车站　　北站　　.694334983825684

- 乘机　乘火车　.576373159885406
- 出差　坐火车　.607399046421051
- 起程　乘火车　.590905547142029
- 搭乘　乘火车　.58670836687088
- 驱车　乘火车　.528952598571777
- 动身　坐火车　.568086743354797

- 火车　　公交车　　.601469516754l5
- 火车　　大巴车　　.578102648258209

- 火车站　　车站　　.7412089l0942078
- 机场　　火车站　　.606023609638214
- 汽车站　　火车站　　.771833002567291
- 火车站　　汽车站　　.771833002567291
- 货运站　　火车站　　.528095364570618
- 火车站　　长途汽车站　　.795943439006805
- 首都机场　　太原火车站　　.540084898471832

- 火车头　　内燃机车　　.460995435714722
- 火车　　列车　　.671974003314972
- 列车　　火车　　.671974003314972
- 火车　　一列　　.589093863964081
- 火车　　铁轨　　.576129078865051

不靠谱的信息

- 火车头　　狼堡　　.480554491281509
- 火车头　　猎鹰　　.43925267457962
- 火车头　　小钢炮　　.448086827993393

- 火车头　　妖人　　.451286911964417
- 航速　　火车皮　　.530428886413574
- 马车　　火车头　　.411905080080032

Word2vec应用

- 可以挖掘词之间的关系，譬如同义词、反义词。
- 可以将词向量作为特征应用到其他机器学习任务中
- 用于机器翻译。分别训练两种语言的词向量，再通过词向量空间中的矩阵变换，将一种语言转变成另一种语言。
- 推理：即已知a之于b犹如c之于d，现在给出 a、b、c，C(a)-C(b)+C(c)约等于C(d)，C(*)表示词向量。可以利用这个特性，提取词语之间的层次关系。
- Connecting Images and Sentences，image understanding。例如文献，DeViSE: A deep visual-semantic em-bedding model。
- Entity completion in Incomplete Knowledge bases or ontologies，即relational extraction。Reasoning with neural tensor net- works for knowledge base completion。
- more word2vec applications，点击link1，link2

- 除了产生词向量，word2vec还有很多其他应用领域，对此我们需要把握两个概念：doc和word。
- 在词向量训练中，doc指的是一篇篇文章，word就是文章中的词。假设我们将一簇簇相似的用户作为doc（譬如QQ群），将单个用户作为word，我们则可以训练user distributed representation，可以借此挖掘相似用户。
- 假设我们将一个个query session作为doc，将query作为word，我们则可以训练query distributed representation，挖掘相似query。

- 思考：你还能想得出来哪类应用可以用word2vec来实现？

操作：

- 1. 打开word2vec_Demo目录
  - 本样例提供了访问两个模型的代码，test_shilu.py访问的是一个在线咨询疾病的模型，test_sogo.py访问的是根据搜狗新闻语料库训练出来的模型。
- 2. 分别运行这两个程序：
  - 横向：输入一个词查询一个词的相关词、输入三个词进行推理分析，输入四个词进行聚类分析
  - 纵向：两者都输入一个相同的词（如“感冒”），了解不同语料对模型的影响。

自动提取关键词、自动摘要、文档相似度计算

- TF-IDF算法
- TextRank算法
- simhash算法
- LSA算法

自动摘要

- 利用计算机将大量的文本进行处理，产生简洁、精炼内容的过程就是文本摘要，人们可通过阅读摘要来把握文本主要内容，这不仅大大节省时间，更提高阅读效率。但人工摘要耗时又耗力，已不能满足日益增长的信息需求，因此借助计算机进行文本处理的自动文摘应运而生。近年来，自动文摘、信息检索、信息过滤、机器识别、等研究已成为了人们关注的热点。

自动文摘（Automatic Summarization）的方法主要有两种：Extraction和Abstraction。

- 其中Extraction是抽取式自动文摘方法，通过提取文档中已存在的关键词，句子形成摘要；Abstraction是生成式自动文摘方法，通过建立抽象的语意表示，使用自然语言生成技术，形成摘要。由于生成式自动摘要方法需要复杂的自然语言理解和生成技术支持，应用领域受限。

主要方法

- **基于统计**：统计词频，位置等信息，计算句子权值，再简选取权值高的句子作为文摘，特点：简单易用，但对词句的使用大多仅停留在表面信息。
- **基于图模型**：构建拓扑结构图，对词句进行排序。例如，TextRank/LexRank
- **基于潜在语义**：使用主题模型，挖掘词句隐藏信息。例如，采用LDA，HMM
- **基于整数规划**：将文摘问题转为整数线性规划，求全局最优解。

TF-IDF算法

- TF-IDF（term frequency–inverse document frequency）是一种用于信息检索与数据挖掘的常用加权技术。TF词频(Term Frequency)，IDF逆向文件频率(Inverse Document Frequency)。
- TF-IDF是一种统计方法，用以评估一字词对于一个文件集或一个语料库中的其中一份文件的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。TF-IDF加权的各种形式常被搜索引擎应用，作为文件与用户查询之间相关程度的度量或评级。除了TF-IDF以外，因特网上的搜索引擎还会使用基于链接分析的评级方法，以确定文件在搜寻结果中出现的顺序。

- IDF =log（D/Dw）其中D是全部网页数，Dw为某个单词出现的次数
- 本质上IDF是一种试图抑制噪音的加权，并且单纯地认为文本频数小的单词就越重要，文本频数大的单词就越无用，显然这并不是完全正确的。IDF的简单结构并不能有效地反映单词的重要程度和特征词的分布情况，使其无法很好地完成对权值调整的功能，所以TFIDF法的精度并不是很高。

词频(tf)、逆文档频率(idf)公式

$$tf-idf=-n*Log(\frac{D_W}{D_T})$$

其中

n：文章中包含该词的个数(tf)
D_W：包括该词的所有文档数
D_T：所有文档数

$$\text{带词权重的}idf = \frac{\sum_{\text{遍历所有包含该词的文档}} (\text{文档中该词的个数} * 1)}{\sum_{\text{遍历所有文档}} (\text{文档中包含的所有词数} * 1)}$$

注：公式中的1是指原本简单累加的文档数

分子就是该词在词库中的词频，即权重公式中的N
分母则是所有词在词库的总词频，即权重公式中的T

将以上分子、分母重新带入$TF-IDF$公式中，即得：

$$W_{tf-idf}=-n*Log(\frac{N}{T})$$

其中：
W_{tf-idf}：引入词权重的$tf-idf$公式
n：文章中包含该词的个数(tf)
$-Log(\frac{N}{T})$：引入词权重的逆文档频率(idf)
N：词库包含该词的个数
T：词库中所有词的个数

TFIDF实现

- 第一步，计算词频

$$\text{词频(TF)} = \text{某个词在文章中的出现次数}$$

- 考虑到文章有长短之分，为了便于不同文章的比较，进行"词频"标准化。

$$\text{词频(TF)} = \frac{\text{某个词在文章中的出现次数}}{\text{文章的总词数}}$$

$$\text{词频(TF)} = \frac{\text{某个词在文章中的出现次数}}{\text{该文出现次数最多的词的出现次数}}$$

- 第二步，计算逆文档频率
- 这时，需要一个语料库（corpus），用来模拟语言的使用环境

$$\text{逆文档频率(IDF)} = \log\left(\frac{\text{语料库的文档总数}}{\text{包含该词的文档数} + 1}\right)$$

- 如果一个词越常见，那么分母就越大，逆文档频率就越小越接近0。分母之所以要加1，是为了避免分母为0（即所有文档都不包含该词）。log表示对得到的值取对数。

- 第三步，计算**TF-IDF**

$$\mathrm{TF-IDF} = \text{词频}(\mathrm{TF}) \times \text{逆文档频率}(\mathrm{IDF})$$

- 可以看到，**TF-IDF**与一个词在文档中的出现次数成正比，与该词在整个语言中的出现次数成反比。所以，自动提取关键词的算法就很清楚了，就是计算出文档的每个词的TF-IDF值，然后按降序排列，取排在最前面的几个词。

TFIDF特点

- TF-IDF算法的优点是简单快速，结果比较符合实际情况。
- 缺点是，单纯以"词频"衡量一个词的重要性，不够全面，有时重要的词可能出现次数并不多。而且，这种算法无法体现词的位置信息，出现位置靠前的词与出现位置靠后的词，都被视为重要性相同，这是不正确的。

TextRank 算法

- 是一种用于文本的基于图的排序算法。其基本思想来源于谷歌的PageRank算法，通过把文本分割成若干组成单元(单词、句子)并建立图模型，利用投票机制对文本中的重要成分进行排序，仅利用单篇文档本身的信息即可实现关键词提取、文摘。和 LDA、HMM 等模型不同，TextRank不需要事先对多篇文档进行学习训练，因其简洁有效而得到广泛应用。

TextRank算法原理

- TextRank算法基于PageRank，用于为文本生成关键字和摘要。
- PageRank最开始用来计算网页的重要性。整个www可以看作一张有向图图，节点是网页。如果网页A存在到网页B的链接，那么有一条从网页A指向网页B的有向边。构造完图后，使用下面的公式：

$$S(V_i) = (1-d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j)$$

- S(Vj)是网页i的中重要性（PR值）。d是阻尼系数，一般设置为0.85。In(Vi)是存在指向网页i的链接的网页集合。Out(Vj)是网页j中的链接存在的链接指向的网页的集合。|Out(Vj)|是集合中元素的个数。

  PageRank需要使用上面的公式多次迭代才能得到结果。初始时，可以设置每个网页的重要性为1。上面公式等号左边计算的结果是迭代后网页i的PR值，等号右边用到的PR值全是迭代前的。

使用TextRank提取关键字

- 将原文本拆分为句子，在每个句子中过滤掉停用词（可选），并只保留指定词性的单词（可选）。由此可以得到句子的集合和单词的集合。
- 每个单词作为pagerank中的一个节点。设定窗口大小为k，假设一个句子依次由下面的单词组成：w1, w2, w3, w4, w5, ..., wn
- w1, w2, ..., wk、w2, w3, ...,wk+1、w3, w4, ...,wk+2等都是一个窗口。在一个窗口中的任两个单词对应的节点之间存在一个无向无权的边。
- 基于上面构成图，可以计算出每个单词节点的重要性。最重要的若干单词可以作为关键词。

使用TextRank提取关键短语

- 参照“使用TextRank提取关键词”提取出若干关键词。若原文本中存在若干个关键词相邻的情况，那么这些关键词可以构成一个关键短语。

  例如，在一篇介绍“支持向量机”的文章中，可以找到三个关键词支持、向量、机，通过关键短语提取，可以得到支持向量机。

使用TextRank提取摘要

- 将每个句子看成图中的一个节点，若两个句子之间有相似性，认为对应的两个节点之间有一个无向有权边，权值是相似度。
- 通过pagerank算法计算得到的重要性最高的若干句子可以当作摘要。

  论文中使用下面的公式计算两个句子Si和Sj的相似度：

$$Similarity(S_i, S_j) = \frac{|\{w_k | w_k \in S_i \& w_k \in S_j\}|}{log(|S_i|) + log(|S_j|)}$$

- 分子是在两个句子中都出现的单词的数量。|Si|是句子i的单词数。

利用simhash算法计算句子/文档相似度

- simhash是一种局部敏感hash。我们都知道什么是hash。那什么叫局部敏感呢，假定A、B具有一定的相似性，在hash之后，仍然能保持这种相似性，就称之为局部敏感hash。
- 通过hash的方法，把上述得到的关键词集合hash成一串二进制，这样我们直接对比二进制数，看其相似性就可以得到两篇文档的相似性，在查看相似性的时候我们采用海明距离，即在对比二进制的时候，我们看其有多少位不同，就称海明距离为多少。

simhash算法

- （1）将文档分词，取一个文章的TF-IDF权重最高的前20个词（feature）和权重（weight）。即一篇文档得到一个长度为20的（feature：weight）的集合。
- （2）对其中的词（feature），进行普通的哈希之后得到一个64为的二进制，得到长度为20的（hash : weight）的集合。
- （3）根据（2）中得到一串二进制数（hash）中相应位置是1是0，对相应位置取正值weight和负值weight。例如一个词进过（2）得到（010111：5）进过步骤（3）之后可以得到列表[-5,5,-5,5,5,5]，即对一个文档，我们可以得到20个长度为64的列表[weight，-weight...weight]。
- （4）对（3）中20个列表进行列向累加得到一个列表。如[-5,5,-5,5,5,5]、[-3,-3,-3,3,-3,3]、[1,-1,-1,1,1,1]进行列向累加得到[-7，1，-9，9，3，9]，这样，我们对一个文档得到，一个长度为64的列表。
- （5）对（4）中得到的列表中每个值进行判断，当为负值的时候去0，正值取1。例如，[-7，1，-9，9，3，9]得到010111，这样，我们就得到一个文档的simhash值了。
- （6）计算相似性。连个simhash取异或，看其中1的个数是否超过3。超过3则判定为不相似，小于等于3则判定为相似。

操作：

- 1. 打开TextRank_Demo目录
- 2. 打开doc目录，查看原始语料。该语料库涉及了不同的几类语料，以展示本示例的功能。
- 3. 打开gettxtinfo.py，了解TextRank的大致用法。
  - 为了使用自己的关键字，我们对TextRank进行了本地化，加入了自己的字典和停用词。
  - 为使摘要更符合原文顺序，我们对摘要的输出顺序按原文进行了调整。
- 4. 运行gettxtinfo.py
- 5. 打开result.txt，观察输出的结果
  - 观察各种不同类型的文本的关键字、关键短语、摘要。
- 思考：是否能用TextRank来帮助自己理解文献？如何结合其他工具快速理解文献？

- 找开simhash目录，查看simhash.py
- 了解下其算法，思考下它可能的应用和存在的问题。

浅层语义分析（LSA）算法

- 是一种自然语言处理中用到的方法，其通过“矢量语义空间”来提取文档与词中的“概念”，进而分析文档与词之间的关系。
- 在某些情况下，LSA又被称作潜在语义索引（LSI）

- LSA的基本假设是，如果两个词多次出现在同一文档中，则这两个词在语义上具有相似性。

- LSA使用大量的文本上构建一个矩阵，这个矩阵的一行代表一个词，一列代表一个文档，矩阵元素代表该词在该文档中出现的次数，然后再此矩阵上使用奇异值分解（SVD）来保留列信息的情况下减少矩阵行数，之后每两个词语的相似性则可以通过其行向量的cos值（或者归一化之后使用向量点乘）来进行标示，此值越接近于1则说明两个词语越相似，越接近于0则说明越不相似。

概述：词-文档矩阵（Occurences Matrix)

- LSA 使用词-文档矩阵来描述一个词语是否在一篇文档中。词-文档矩阵式一个稀疏矩阵，其行代表词语，其列代表文档。一般情况下，词-文档矩阵的元素是该词在文档中的出现次数，也可以是是该词语的tf-idf(term frequency–inverse document frequency)。
- 词-文档矩阵和传统的语义模型相比并没有实质上的区别，只是因为传统的语义模型并不是使用“矩阵”这种数学语言来进行描述。

降维

- 在构建好词-文档矩阵之后，LSA将对该矩阵进行降维，来找到词-文档矩阵的一个低阶近似。降维的原因有以下几点：
  - 原始的词-文档矩阵太大导致计算机无法处理，从此角度来看，降维后的新矩阵式原有矩阵的一个近似。
  - 原始的词-文档矩阵中有噪音，从此角度来看，降维后的新矩阵式原矩阵的一个去噪矩阵。
  - 原始的词-文档矩阵过于稀疏。原始的词-文档矩阵精确的反映了每个词是否“出现”于某篇文档的情况，然而我们往往对某篇文档“相关”的所有词更感兴趣，因此我们需要发掘一个词的各种同义词的情况。

推导

- 假设X是词-文档矩阵，其元素（i,j）代表词语i在文档j中的出现次数，则X矩阵看上去是如下的样子：

$$\begin{array}{c} \\ \mathbf{t}_i^T \rightarrow \end{array} \begin{array}{c} \mathbf{d}_j \\ \downarrow \\ \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix} \end{array}$$

- 可以看到，每一行代表一个词的向量，该向量描述了该词和所有文档的关系。

- 可以看到，每一行代表一个词的向量，该向量描述了该词和所有文档的关系。

$$\mathbf{t}_i^T = \begin{bmatrix} x_{i,1} & \dots & x_{i,n} \end{bmatrix}$$

- 同理，一列代表一个文档向量，该向量描述了该文档与所有词的关系：

$$\mathbf{d}_j = \begin{bmatrix} x_{1,j} \\ \vdots \\ x_{m,j} \end{bmatrix}$$

- 词向量$t_i^T t_p$的点乘可以表示这两个单词在文档集合中的相似性。矩阵XX^{T}包含所有词向量点乘的结果，元素（i,p）和元素（p,i）具有相同的值，代表词p和词i的相似度。类似的，矩阵$X^T X$包含所有文档向量点乘的结果，也就包含了所有文档那个的相似度。

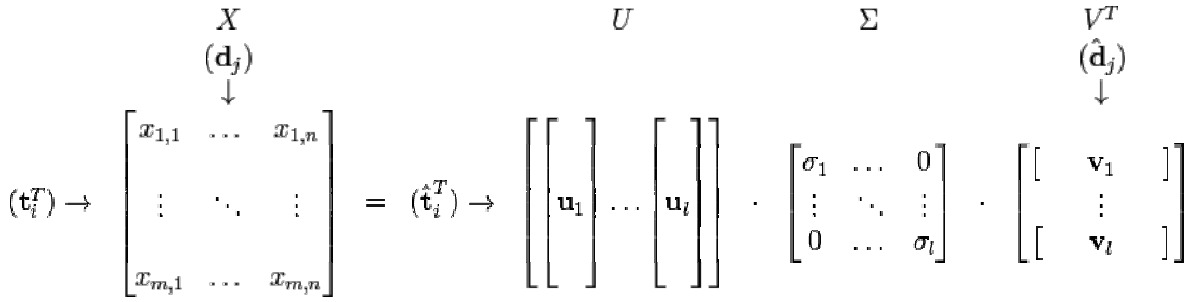
- 现在假设存在矩阵X的一个分解，即矩阵X可分解成正交矩阵U和V，和对角矩阵∑的乘积，这种分解叫做奇异值分解（SVD），即：

$$X = U\Sigma V^T$$

- 因此，词与文本的相关性矩阵可以表示为：

$$\begin{aligned} XX^T &= (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)(V^{T^T}\Sigma^T U^T) = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T \\ X^T X &= (U\Sigma V^T)^T(U\Sigma V^T) = (V^{T^T}\Sigma^T U^T)(U\Sigma V^T) = V\Sigma^T U^T U\Sigma V^T = V\Sigma^T\Sigma V^T \end{aligned}$$

- 因为$\Sigma\Sigma^T$与$\Sigma^T\Sigma$是对角矩阵，因此U 肯定是由XX^T的特征向量组成的矩阵，同理V是$X^T X$特征向量组成的矩阵。这些特征向量对应的特征值即为\Sigma \Sigma^T中的元素。综上所述，这个分解看起来是如下的样子：



- 上述公式可以变换为：
  $X_k = U_k \Sigma_k V_k^T$
- 有了这个变换，则可以做以下事情：
  - 判断文档 j 与 q 在低维空间的相似度。
  - 判断词i和词p的相似度。
  - 有了相似度则可以对文本和文档进行聚类。
  - 给定一个查询字符串，算其在语义空间内和已有文档的相似性。

- 要比较查询字符串与已有文档的相似性，需要把文档和查询字符串都映射到语义空间，对于原始文档，由以下公式可以进行映射：

$$\hat{\mathbf{d}}_j = \Sigma_k^{-1} U_k^T \mathbf{d}_j$$

- 同理，对于查询字符串，得到其对应词的向量后，根据公式:

$$\hat{\mathbf{q}} = \Sigma_k^{-1} U_k^T \mathbf{q}$$

- 将其映射到语义空间，再与文档进行比较。
- 低维的语义空间可以用于以下几个方面:
  - 在低维语义空间可对文档进行比较，进而可用于文档聚类和文档分类。
  - 在翻译好的文档上进行训练，可以发现不同语言的相似文档，可用于跨语言检索。
  - 发现词与词之间的关系，可用于同义词、歧义词检测。d
  - 通过查询映射到语义空间，可进行信息检索。
  - 从语义的角度发现词语的相关性，可用于“选择题回答模型”

LSA的一些缺点

- 新生成的矩阵的解释性比较差。比如
  - {(car), (truck), (flower)} ↦ {(1.3452 * car + 0.2828 * truck), (flower)}
  - (1.3452 * car + 0.2828 * truck) 可以解释成 "vehicle"。同时，也有如下的变换
  - {(car), (bottle), (flower)} ↦ {(1.3452 * car + 0.2828 * bottle), (flower)}
  - 造成这种难以解释的结果是因为SVD只是一种数学变换，并无法对应成现实中的概念。

- LSA无法扑捉一词多以的现象。在原始词-向量矩阵中，每个文档的每个词只能有一个含义。比如同一篇文章中的“The Chair of Board"和"the chair maker"的chair会被认为一样。在语义空间中，含有一词多意现象的词其向量会呈现多个语义的平均。相应的，如果有其中一个含义出现的特别频繁，则语义向量会向其倾斜。
- LSA具有词袋模型的缺点，即在一篇文章，或者一个句子中忽略词语的先后顺序。
- LSA的概率模型假设文档和词的分布是服从联合正态分布的，但从观测数据来看是服从泊松分布的。因此LSA算法的一个改进PLSA使用了多项分布，其效果要好于LSA。

参考

- TextRank
- TextRank4ZH
- snownlp

操作：

- 1. 打开LSI_Demo目录
- 2. 打开QA_all.txt，了解语料格式
- 3. 打开test.py，修改问题
  - 注意示例提供了四种查询：一种是常识查询，二是业务查询，三是分诊信息查询，四是图灵机器人查询。
- 4. 运行test.py
- 5. 打开result.txt，观察输出的结果

谢谢大家！

刘翔 scsdlx@outlook.com