第9讲 语义分析
semantics analysis

Language without meaning is meaningless!

语义

- 词的语义和句子的语义：
  - 有意义词、无意义词
  - 有意义的句子和无意义的句子
  - 词的歧义
    - **bear**
  - 句子的歧义
    - **Jack saw a man with a telescope.**
    - 咬死了猎人的狗。
    - 开刀的是他父亲。

语义基本概念及计算

- 语义基本概念：
  - 义位
  - 语义场
  - 语义特征
  - 格语法
  - 语义网络
- 语义计算：
  - 词与词之间的语义相似度的计算
  - 句子之间的语义
  - 篇章之间的语义

语义概念--义 位

- 在词典编撰中，称每一个词义为一个义项，在语义学中也称之为义位：
  - 如：“明白”有4个不同的意思：
    - 内容、意思等使人容易了解；清楚；明确；
    - 公开的、不含糊的；
    - 聪明；懂道理；
    - 知道；了解
  - 这表明“明白”这个词包含四个不同的义位.

义位间的关系或定义

同形/同音异义词

1. **Homographs(同形)**
   - **$bank_1$: financial institution, $bank_2$: sloping land**
   - **bat_1: club for hitting a ball, bat_2: nocturnal flying mammal(夜间飞行动物)**
2. **Homophones(同音):**
   - **Write and right**
   - **Piece and peace**

同形/同音异义词所导致的问题

- 信息检索
  - 苹果 手机
- 机器 翻译
  - I can‘t bear you.
- 语音识别
  - 报复/抱负

多义词(Polysemy)

- 有关bank的两个句子，表达两个不同意思：
  - 1. The bank was constructed in 1875 out of local red brick.
  - 2. I withdrew the money from the bank

异形同义词

- 词不同但在不同上下文下，表示的意思相同：
  - **couch / sofa**
  - **big / large**
  - **automobile / car**
  - **Water / H_20**

异形同义词反应的是某个义位上的同义!

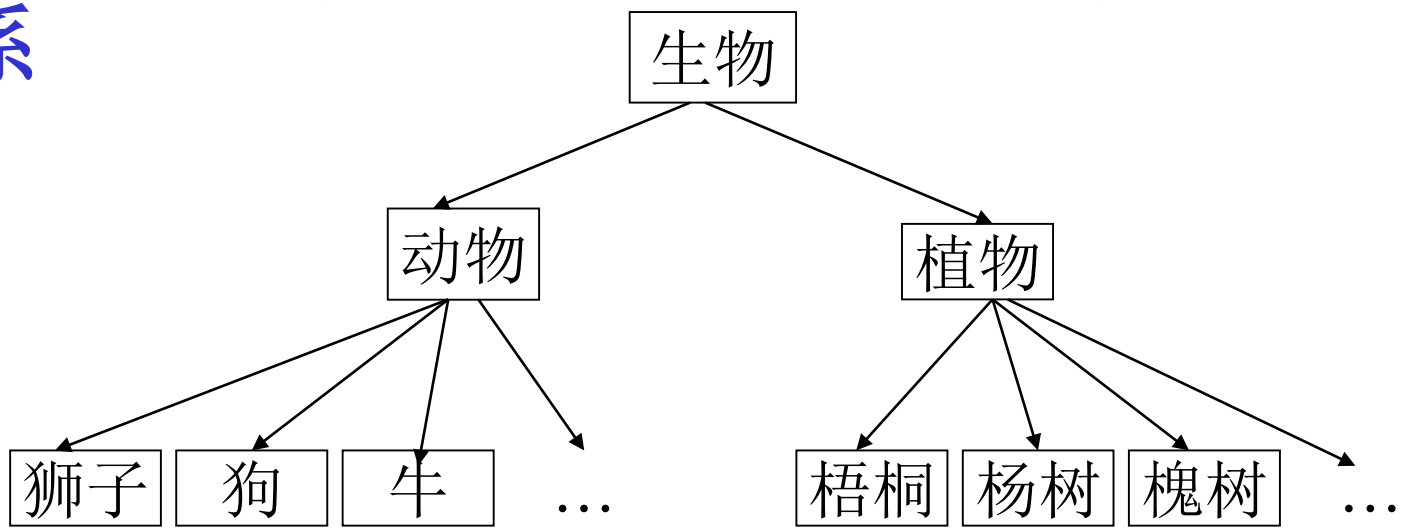
反义词

dark/light short/long fast/slow rise/fall

hot/cold up/down in/out

上义词和下义词(上下义位关系)

- 指两个义位(上义义位和下义义位)间存在类属关系



- 狮子和狗是同位关系(co-hyponyms)
- 杨树是植物的下位关系词(hyponym)
- 生物是动物和植物的上义词(hypernymy)

例子

WordNet 英英词典

desk

desks

Noun

- a piece of furniture with a writing surface and usually drawers or other compartments

  (hypernym) table

  (hyponym) davenport 长椅

  (part-meronym) drawer

整体-部分关系 part-meronym

- 一个义位所表达的对象是另一个义位所表达的对象的组成部分。
  - 例如：手是身体的一部分；
  - **body, arm**
  - **house, roof**

整体-部分关系 例

WordNet 英英词典

dog

dogs dogged dogging

Noun

- a member of the genus Canis (probably descended from the common wolf) domesticated by man since prehistoric times; occurs in many breeds; "the d night"

  (synonym) domestic dog, Canis familiaris

  (hypernym) canine, canid 犬科动物

  (hyponym) pooch, doggie, doggy, barker, bow-wow

  (member-holonym) Canis, genus Canis

  (part-meronym) flag

WordNet中单词“bass”的义位层次

- S: (n) **bass**, basso (an adult male singer with the lowest voice)
  - *direct hypernym* / ***inherited hypernym*** / *sister term*
    - S: (n) singer, vocalist, vocalizer, vocaliser (a person who sings)
      - S: (n) musician, instrumentalist, player (someone who plays a musical instrument (as a profession))
        - S: (n) performer, performing artist (an entertainer who performs a dramatic or musical work for an audience)
          - S: (n) entertainer (a person who tries to please or amuse)
            - S: (n) person, individual, someone, somebody, mortal, soul (a human being) *"there was too much for one person to do"*
              - S: (n) organism, being (a living thing that has (or can develop) the ability to act or function independently)
                - S: (n) living thing, animate thing (a living (or once living) entity)
                  - S: (n) whole, unit (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"; "the team is a unit"*
                    - S: (n) object, physical object (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
                      - S: (n) physical entity (an entity that has physical existence)
                        - S: (n) entity (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

WordNet中名词关系

Relation	Also called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	*breakfast*^{1} → *meal*^{1}
Hyponym	Subordinate	From concepts to subtypes	*meal*^{1} → *lunch*^{1}
Member Meronym	Has-Member	From groups to their members	*faculty*^{2} → *professor*^{1}
Has-Instance		From concepts to instances of the concept	*composer*^{1} → *Bach*^{1}
Instance		From instances to their concepts	*Austen*^{1} → *author*^{1}
Member Holonym	Member-Of	From members to their groups	*copilot*^{1} → *crew*^{1}
Part Meronym	Has-Part	From wholes to parts	*table*^{2} → *leg*^{3}
Part Holonym	Part-Of	From parts to wholes	*course*^{7} → *meal*^{1}
Antonym		Opposites	*leader*^{1} → *follower*^{1}

WordNet 3.0

- 网址:
  - http://wordnetweb.princeton.edu/perl/webwn
- **Wordnet可以在NLTK中导入**
  - http://www.nltk.org/Home

WordNet 3.0统计信息

POS	Unique Strings	Synsets
Noun	117798	82115
Verb	11529	13767
Adjective	21479	18156
Adverb	4481	3621
Totals	155287	117659

--统计来自wordnet官网

WordNet 3.0统计信息

POS	Monosemous (单义词)	Polysemous (多义词)	Polysemous (多义词)
	Words and Senses	**Words**	**Senses**
Noun	101863	15935	44449
Verb	6277	5252	18770
Adjective	16503	4976	14399
Adverb	3748	733	1832
Totals	**128391**	**26896**	**79450**

--统计来自wordnet官网

WordNet 3.0统计信息

POS	Average Polysemy Including Monosemous Words	Average Polysemy Excluding Monosemous Words
Noun	1.24	2.79
Verb	2.17	3.57
Adjective	1.4	2.71
Adverb	1.25	2.5

--统计来自wordnet官网

WordNet的应用例

- 可通过句子内部概念密度的计算，消除一部分歧义
  - 例如有一个句子：
    - **I withdrew the money from the bank .**
  - 几个问题：
    - **bank一共有几个义项？**
    - **如何判断该句子中的bank是取“银行”还是“堤岸”，还是其他义项的意思？**

WordNet的应用例-问题1

- bank的义项一共有：
  - 名词有10个义项
  - 动词有7个义项
  - bank的义项

WordNet的应用例-问题2

- 如何判断该句子中的bank是取“银行”还是“堤岸”，还是其他义项的意思？
  - 分别计算“银行”和“堤岸”与“money”的相似度
    - **sim(bank-1,money-1)=0.0573004**
    - **sim(bank-2,money-1)=0.0575236**
    - **这里bank-1为银行义项，bank-2为堤岸义项。**

WordNet的应用

- **WORDNET在计算语言界备受热衷**
  - 如：依靠**wordnet**名词的语义消歧可超过**60%**
  - 它被广泛应用于主题含义识别；图像检索；文本语义分类；网上文本过滤；语料库语义标注等方面

http://wordnet.princeton.edu

知网（Hownet）

- 网站：http://www.keenage.com
- 概念描述举例
- NO.=017144
  - **W_C=打**
  - **G_C=V**
  - **E_C=~网球，~牌，~秋千，~太极，球~得很棒**
  - **W_E=play**
  - **DEF=exercise|锻练,sport|体育**
  - **其中DEF是核心，采用特定的“知识描述语言”**

知网（Hownet）

- 打017144　exercise|锻练,sport|体育
- 男人059349 human|人,family|家,male|男
- 生日072280 time|时间,day|日,@ComeToWorld|问世,$congratulate|祝贺
- 写信089834 write|写,ContentProduct=letter|信件
- 北京003815 place|地方,capital|国都,ProperName|专,(China|中国)
- 儿童基金会024083 part|部件,%institution|机构,politics|政,#young|幼,#fund|资金,(institution|机构=UN|联合国)

知网（Hownet）

- 义原总数：1500多个
- 义原分类：共8类
  - – 基本义原
  - 事件、实体、次要特征
  - 属性、属性值、数量、数量值
- – 语法义原：描述语法特征，如POS
  - 语法
- – 关系义原：描述意义关系，类似于格关系
  - 动态角色
  - 动态属性

语义分析的作用

- 基于语义的摘要
- 基于语义的情感分析(同义词、近义词等等)
- 基于语义情感分析的音乐推荐等
- …… ……

格语法

- 菲尔摩(C.J.Fillmore)在题为《格辩》的论文中，提出了格语法

- 格语法中，利用句子的动词周围的名词性成分与动词的语义组合关系来形成表达句子意义的格结构。

格语法

格语法是美国语言学家菲尔摩于1966年提出的一种新理论。

他认为：句法分析中的主语、宾语等语法关系只是表层结构上的概念，在语言的底层，所需要的不是这些表层的语法关系，而是用施事、受事、工具、受益等概念所表示的句法语义关系。

而这些语义关系经过变换后，才在表层结构中成为主语或宾语。

格的含义

- “格”-case：原指某些屈折语中用于表示词间语法关系的名词和代词的形态变化。如主格、宾格等--这些是传统上的格，属于表层格。

- 格语法中的格是“深层格”，是指句子中词与词之间的及物性关系，如：动作和施事者的关系、动作和受事者的关系等，这些关系就是语义关系。

最初列出的6个格：

- 施事格(Agentive)
  - 句子主动词所表现的事件、行为或状态等的主动发起者，如：**Tom broke the windows.**
- 工具格(Instrumental)
  - 该成分代表的对象是句子主动词所表现的事件、行为中使用的工具，如**Tom broke the windows with a ball.**
- 与格(Dative)
  - 该成分代表的对象是句子主动词所表现的事件、行为的参与者，如**Tome give me a ball.**

最初列出的6个格：

- 使成格(Factitive)
  - 由动词确定的动作或状态所形成的客体
  - 如： **John dreamed a dream about Mary.**
- 方位格(locative)
  - 表示由动词或状态的处所或空间方位
  - 如. **He is in the house.**
- 客体格(objective)
  - 表示由动词确定的动作或状态所影响的事物
  - 如. **He bought a book.**

格语法

- 菲尔摩的格也被称为语义角色，深层格等。
- 格体现了句子动词和名词的语义组合关系。

举 例

- 看下面几个例子：

(1) The door opened.
(2) The key opened the door.
(3) The boy opened the door.
(4) The door was opened by the boy.
(5) The boy opened the door with a key.

- 分析以上各句：
  - “表层结构”不同：
    - 各句的语法结构有所不同；
    - 主语、谓语等属性也有所不同；
  - “深层结构”却是一致的：
    - 施事格：**the boy**
    - 客体格：**the door**（也称受事格）
    - 工具格：**the key**
  - 均是针对动词“**open**”的语义关系。

格语法

- 格语法通常有三部分组成：
  - 基本规则
  - 词汇部分
  - 转换部分

基本规则

(1) S → M(形态)+P(命题)

形态→时、态、句式、情态和时间等

(2) P → V+C1+C2+…Cn

P→Vb+格变元

Vb → run, walk, break,…

(3) C → K+NP

格变元→格关系+[NP|S]

格关系→AGT,OBJ,SOUR,LOC,TIME…

格表

- 菲尔摩认为命题中需要的格包括：
  - 施事格：**he laguhed.**
  - 工具格：**he cut the rope with a knife.**
  - 承受格　**he is tall.**
  - 使成格　**John dreamed a dream about Mary.**
  - 方位格　**He is in the house.**
  - 客体格　**He bought a book.**
  - 受益格　**He sang a song for Mary.**
  - 源点格　**I bought a book from Mary.**

格表

- 终点格：I sold a car to Mary.
- 伴随格：he sang a song with Mary.

底层格是格语法解释语义和句法现象的基本工具。但确定一张完整的格的清单却十分困难。

表 2： FrameNet 框架示例：Removing

框架名	Removing（移开）	
框架描述	An **Agent** causes a **Theme** to move away from a location, the **Source**.	
框架元素	Agent 施事	The Agent is the person (or other force) that causes the Theme to move.
	Cause 致事	The noise of impact resulting from caused-motion of a Theme
	Theme 当事	Theme is the object that changes location.
	Cotheme 同事	The Cotheme is the second moving object, expressed as a direct object.
	Distance 距离	The Distance is any expression which characterizes the extent of motion.
	Goal 目标	The Goal is the location where the Theme ends up.
	Path 路径	Path along which moving occurs.
	Result 结果	Result of an event
	Source 起点	The initial location of the Theme, before it changes location.
	Vehicle 交通工具	The means of conveyance controlled by the Driver.
词例	abduct.v, clear.v, confiscate.v, depose.v, discard.v, dislodge.v, drain.v, eject.v, ejection.n, eliminate.v, elimination.n, empty.v, evacuate.v, evacuation.n, evict.v, eviction.n, ...	

使用格语法进行语义分析

- 分析的结果可用“格框架”来表示

如：**In the room,he broke a window with a hammer.**

格框架：**[BREAK**

[case-frame

agentive:he

objective:window

instrumental:hammer

locative: room

[MODALS:

time: past]

使用格语法进行语义分析

- 具体分析抽象步骤：

(1)判断待分析词序列中主要动词，并在动词词典中找出该动词的格框架。

(2)对格内容进行相应的填充。

(3)根据句子中出现的标志判断句子的情态Modal.

例

- 分析句子 **The young athlete will be running in Los Angeles next week.**
  - 利用句法分析，找到**NP: the young athlete,** 压入堆栈
  - 利用**will** 找到句子的**主要动词run**，且时态为将来时
  - **NP: In Los Angeles** ，因为在**VP**后，且**Los Angeles** 为地名，及**in**，可以判定该**NP**为地点格
  - **NP: next week.**

例 (续)

- 从动词词典中查run的格框架,如：
  - **Verb : run**
  - **Case Frame**

    [Neutral --required

    Dative --not allowed

    Locative –optical

    Instrumental –not allowed

    Agentive --required]

例 (续)

- 此时，有2个NP，下面依次填充格框架
  - 对中性格, 由于run的中性格要求为实体或组织，因此取the young athlete
  - 对施事格，依据run的要求，取the young athlete
  - 时态等也已经确定

例 (续)

- 填充完后得到该句子的格框架如下：

CASE

[Agentive: the young athlete

Locative: in Los Angeles

Neutral: the young athlete]

[Modal

Tense(时): future

Aspect(态): Perfect

MOOD(句式) Declarative

Essence(性): Positive

Time: next week]

格语法描写汉语的局限性

- 汉语中的流水句、无动词、省略等结构，无法或不必用一个动词统帅一个句子的模式来描述。

格语法在机器翻译中的应用-例1

In the room,he broke a window with a hammer.

原语言句格框架：	目标语言格框架：
[BREAK	[打破
[case-frame	[case-frame]
agentive:he	施事:他
objective:window	受事:窗户
instrumental:hammer	工具:斧子
locative: room	地点:房间
[MODALS:	[时态：
time: past]	时间：过去
式]	

格语法在机器翻译中的应用-例2

“I am reading the book in the school” ← “我在学校读书”

目标语言格框架：

[read
  [case-frame
    agentive: I
    objective:book
    instrumental:
    locative: school
  [MODALS:
    time:]

源语言格框架：

[读
  [case-frame]
    施事:我
    受事:书
    工具:
    地点:学校
  [时态:
  时间：进行式]

词的相似度计算

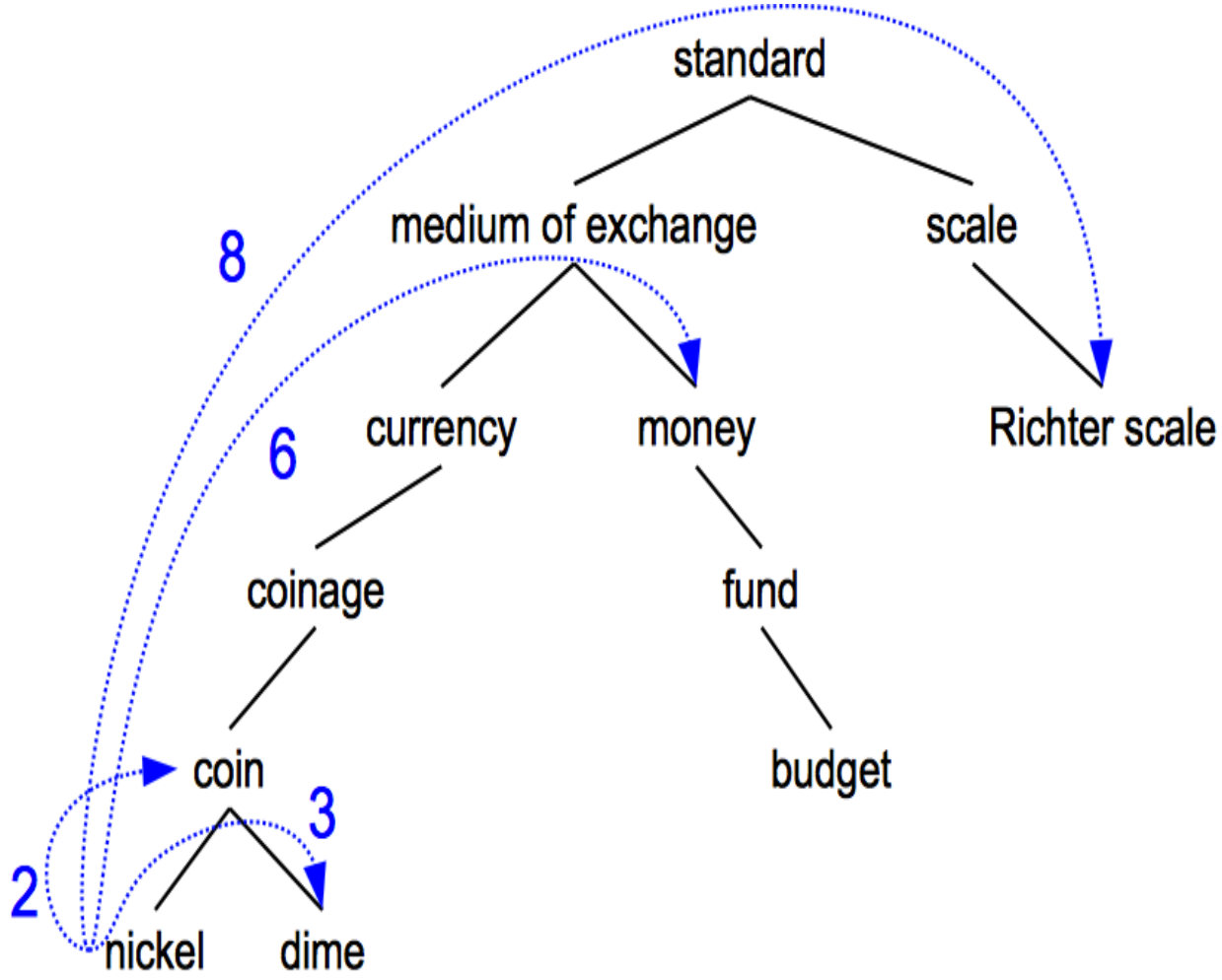
词相似度计算的应用

- 信息检索
- 自动问答
- 机器翻译
- 作文自动评分
- 抄袭检测
- 文档聚类
- …… ……

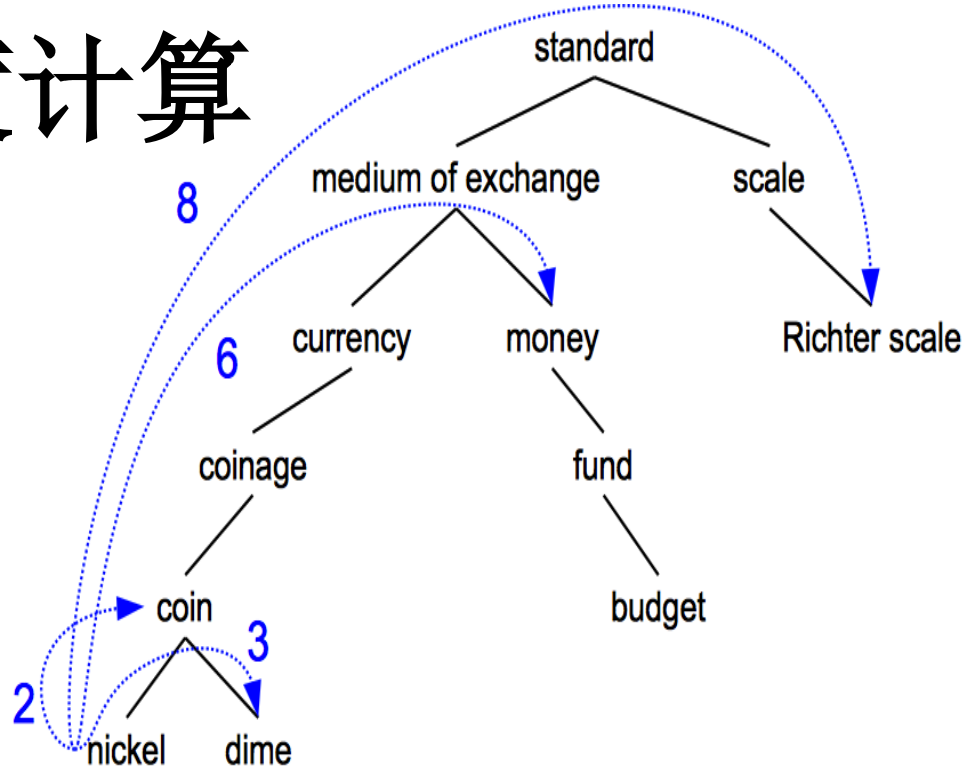
两种方法

- 基于词典的方法
  - 在词典树上是否靠近？
- 基于统计的方法
  - 是否具有相似的上下文?
- 基于深度学习的方法
  - **Word2Vec**

同义词词典树


standard
medium of exchange
scale
8
currency
money
Richter scale
6
coinage
fund
coin
budget
3
2
nickel
dime

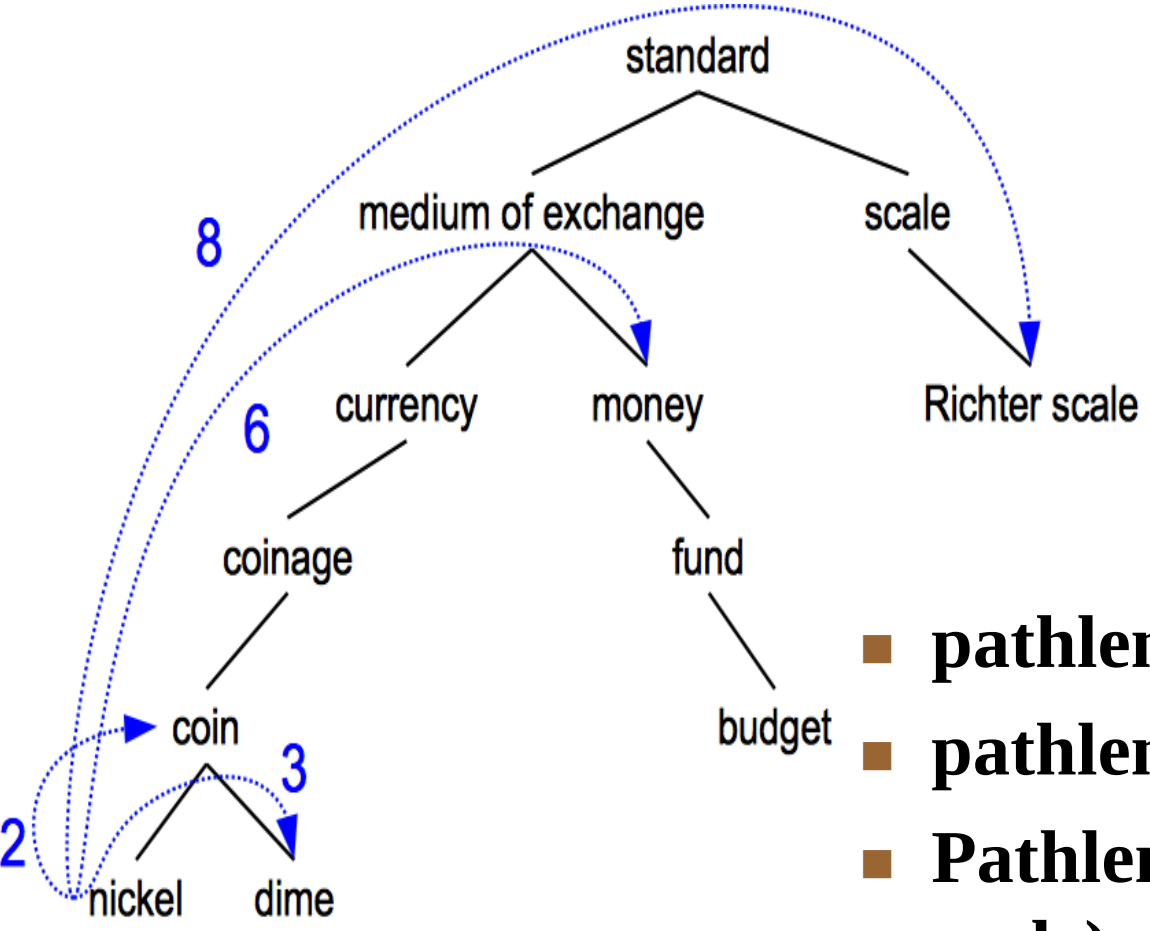
基于路径的相似度计算



- 两个义项的相似度决定于他们的最短路径

基于路径的相似度计算方法

pathlen(c_1,c_2) = 1 + 边个数

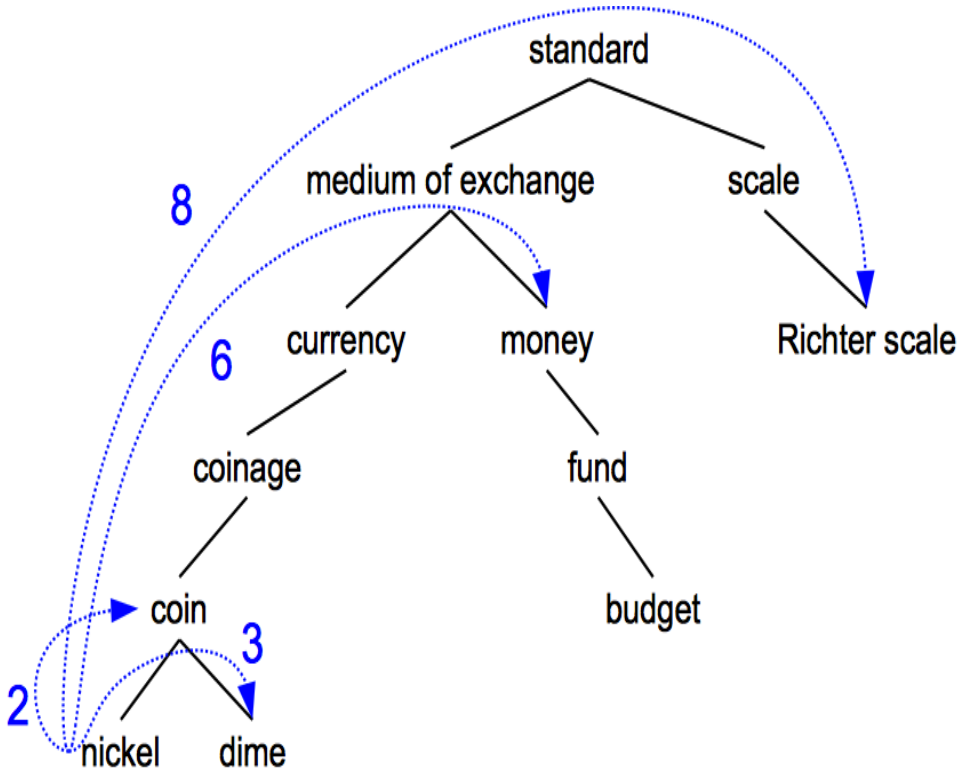


- **pathlen(nicckel, dime)=?**
- **pathlen(coin, fund)=?**
- **Pathlen(budgetn, richter scale)=?**

基于路径的相似度计算方法

- 两个义项c1,c2之间的路径相似度被定义为:
- **simpath(c_1,c_2)** = $\dfrac{1}{\text{pathlen}(c_1, c_2)}$
- 两个单词w1,w2之间的相似度被定义为:
- **wordsim(w_1,w_2)** = $\max_{c_1 \in \text{senses}(w_1), c_2 \in \text{senses}(w_2)} \text{sim}(c_1, c_2)$

Example:



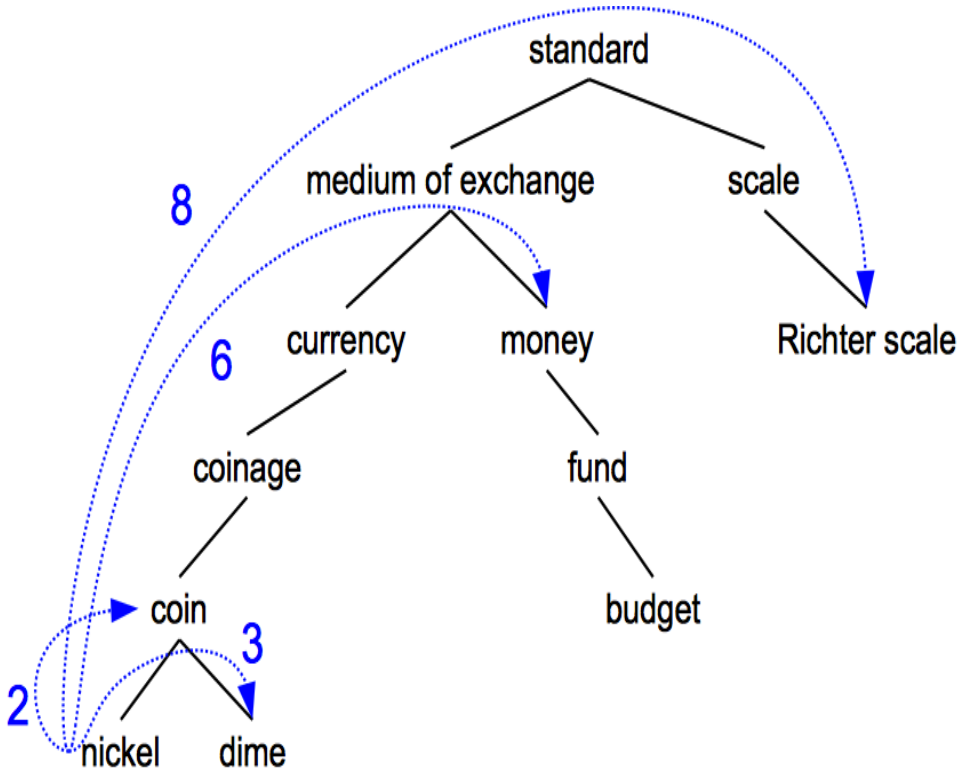
simpath(*fund,budget*) = 1/2 = .5

simpath(*nickel,currency*) = ?

simpath(*nickel,money*) = ?

simpath(*coinage,Richter scale*) =?

Example:



假定单词A有2个意思:(coin, money)

单词B有3个意思: (scale, standand, fund)

计算单词A与B之间的相似度

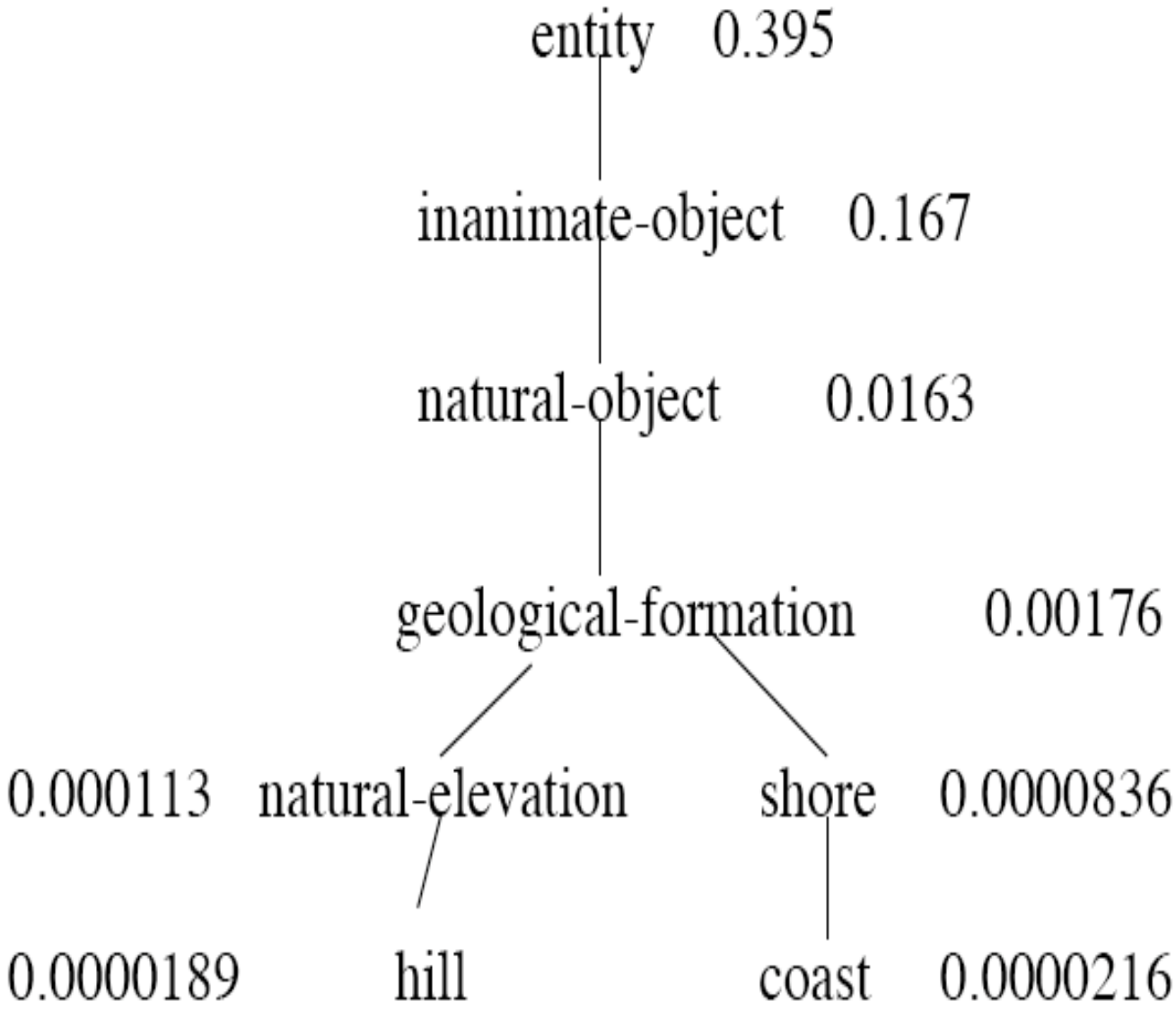
Resnik 1995. Using information content to evaluate semantic similarity in a taxonomy. IJCAI

- 给字典树中每个节点计算相应的概率P(c)，计算公式如下：

$$P(c) = \frac{\sum_{w \in words(c)} count(w)}{N}$$

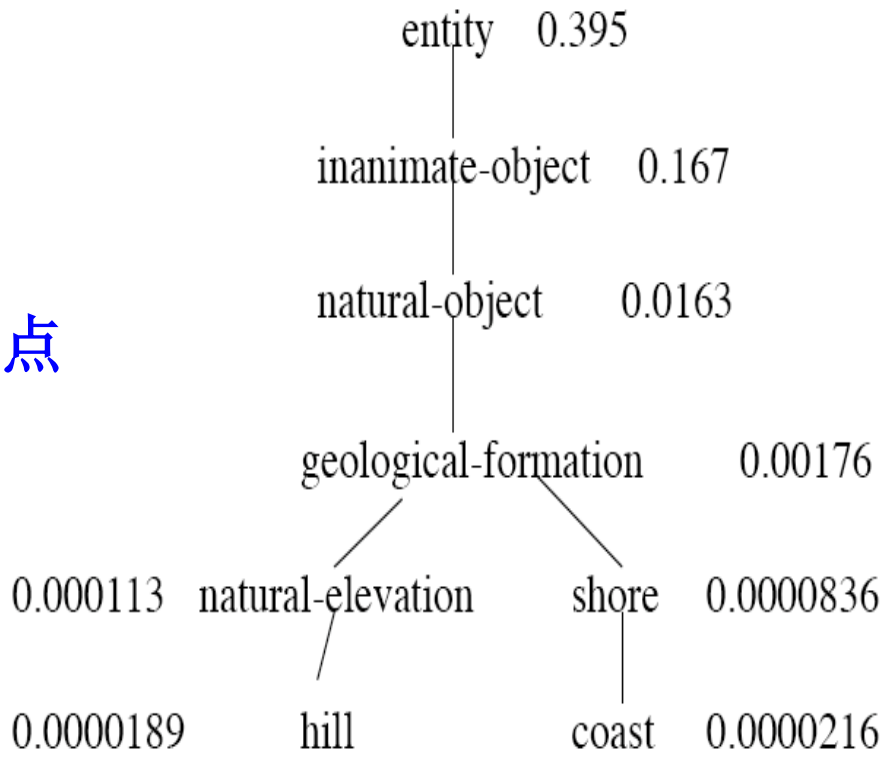
分子为c的所有子节点

分母为语料库中所有单词个数


entity 0.395
inanimate-object 0.167
natural-object 0.0163
geological-formation 0.00176
0.000113 natural-elevation
shore 0.0000836
0.0000189 hill
coast 0.0000216

新的定义：

- **Information content:**
  IC(c)= -log P(c)
- **Lowest common subsumer**
  LCS(c_1,c_2)= 包含c1和c2的最小节点



- LCS(hill,coast)=?

Geological-formation

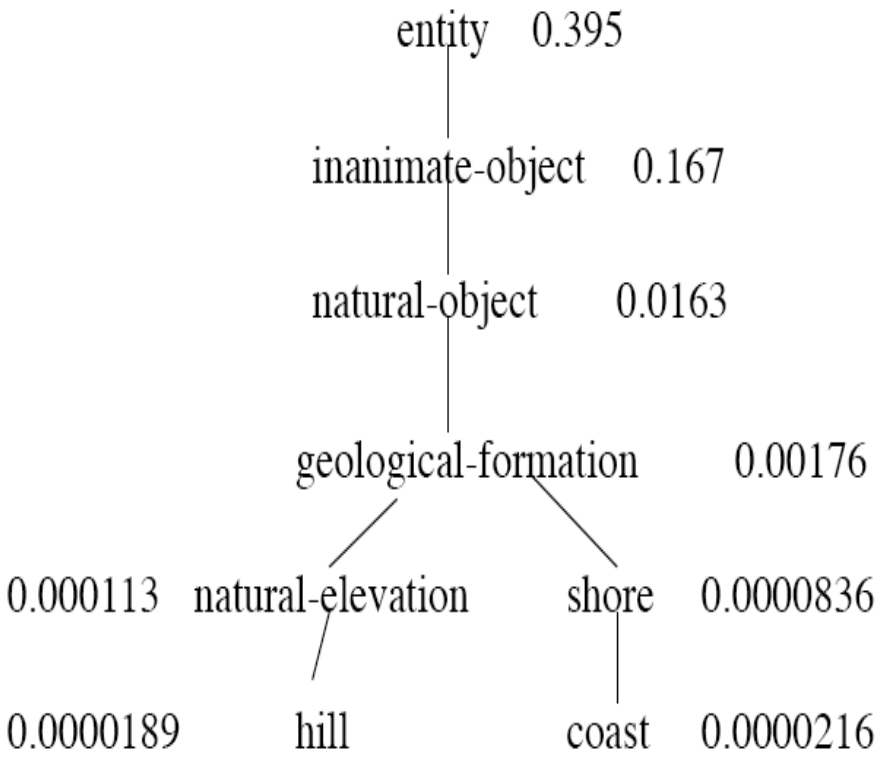
Resnik 相似度计算方法

- 两个词的相似度和他们的共有信息相关
- 两个词的共有信息越多，两个词越相似
- **Resnik**相似度计算公式为：

$$sim_{resnik}(c_1,c_2) = -\log P(LCS(c_1,c_2))$$

Example --Resnik method

- $\mathbf{sim}_{\mathbf{resnik}}$(hill,coast)=?



Dekang Lin. 1998. An Information-Theoretic Definition of Similarity. ICML

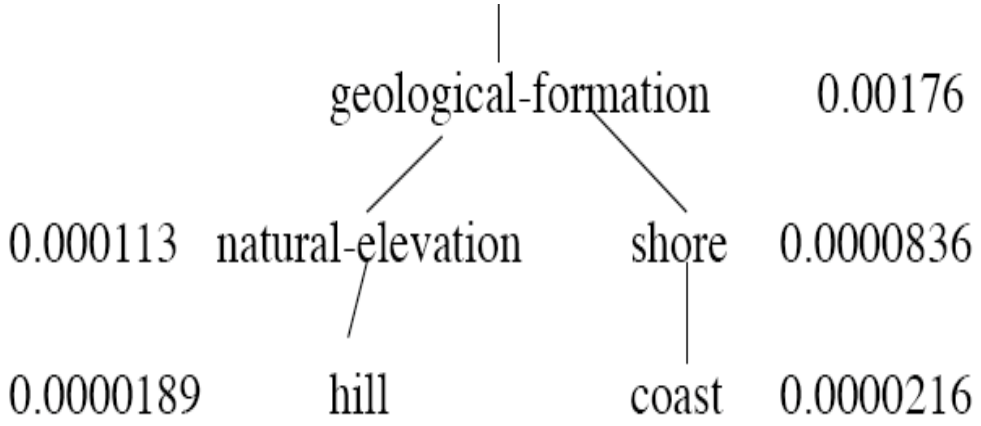
- 两个词共有信息越多，则越相似；
- 两个词差异信息越少，则越相似；
- 共有信息: IC(common(A,B))
- 差异信息: IC(description(A,B)-IC(common(A,B))

Dekang Lin. 1998. An Information-Theoretic Definition of Similarity. ICML

$$sim_{Lin}(A,B) \propto \frac{IC(common(A,B))}{IC(description(A,B))}$$

$$sim_{Lin}(c_1,c_2) = \frac{2\log P(LCS(c_1,c_2))}{\log P(c_1)+\log P(c_2)}$$

例：



$$sim_{Lin}(A,B) = \frac{2\log P(LCS(c_1,c_2))}{\log P(c_1)+\log P(c_2)}$$

$$sim_{Lin}(\text{hill},\text{coast}) = \frac{2\log P(\text{geological-formation})}{\log P(\text{hill})+\log P(\text{coast})}$$

$$= \frac{2\ln 0.00176}{\ln 0.0000189 + \ln 0.0000216}$$

$$= .59$$

基于词典的词相似度方法

$$\mathrm{sim}_{\mathrm{path}}(c_1,c_2)=\frac{1}{pathlen(c_1,c_2)}$$

$$\mathrm{sim}_{\mathrm{resnik}}(c_1,c_2)=-\log P(LCS(c_1,c_2)) \quad \mathrm{sim}_{\mathrm{lin}}(c_1,c_2)=\frac{2\log P(LCS(c_1,c_2))}{\log P(c_1)+\log P(c_2)}$$

$$\mathrm{sim}_{\mathrm{jiangconrath}}(c_1,c_2)=\frac{1}{\log P(c_1)+\log P(c_2)-2\log P(LCS(c_1,c_2))}$$

$$\mathrm{sim}_{eLesk}(c_1,c_2)=\sum_{r,q\in RELS}\mathrm{overlap}(\mathrm{gloss}(r(c_1)),\mathrm{gloss}(q(c_2)))$$

Libraries for computing thesaurus-based similarity

- **NLTK**
  - http://nltk.github.com/api/nltk.corpus.reader.html?highlight=similarity - nltk.corpus.reader.WordNetCorpusReader.res_similarity

- **WordNet::Similarity**
  - http://wn-similarity.sourceforge.net/
  - **Web-based interface:**
    - http://marimba.d.umn.edu/cgi-bin/similarity/similarity.cgi

基于统计的词相似度计算

基于词典的方法的问题

- 是否有足够大的词典？
  - 词典中词缺失
  - 某些义项之间的关系缺失
  - 词典中对动词和形容词效果不佳
    - 形容词和动词不如名词，相对而言，具有较少的上下位关系

基于统计的方法

- **example:**

  A bottle of *tesgüino* is on the table
  Everybody likes *tesgüino*
  Tesgüino makes you drunk
  We make *tesgüino* out of corn.

- **从词的上下文，可以猜测*tesgüino* 的意思**
  - **两个词如果具有相似的上下文，则语义相近**

词-文档矩阵

- 每个文档用一个向量表示
- 矩阵中的值为词t在文档d中出现的次数$tf_{t,d}$:

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	6	117	0	0

词-文档矩阵

- 两个文档相似，如果他们的向量相似

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	6	117	0	0

词-文档矩阵

- 每个词可以看成是横的向量

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	6	117	0	0

词-文档矩阵

- 两个词向量相似，则两个词的语义相似

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	6	117	0	0

词-上下文矩阵

- 可以不使用整个文档，而是使用上下文信息：
  - **Paragraph**
  - **Window of 10 words**
- 词可以定义为包含上下文词数的向量

例如: 20个词的上下文 (来自布朗语料库)

- **equal amount of sugar, a sliced lemon, a tablespoonful of apricot preserve or jam, a pinch each of clove and nutmeg,**
- **on board for their enjoyment. Cautiously she sampled her first pineapple and another fruit whose taste she likened to that of**

- of a recursive type well suited to programming on the **digital** computer. In finding the optimal R-stage policy from that of
- substantially affect commerce, for the purpose of gathering data and **information** necessary for the study authorized in the first section of this

词-上下文矩阵

- 两个词的上下文向量如果相似，则这两个词相似

	aardvark	computer	data	pinch	result	sugar	...
apricot	0	0	0	1	0	1	
pineapple	0	0	0	1	0	1	
digital	0	2	1	0	1	0	
information	0	1	6	0	4	0	

PMI值的应用

- 为了避免词向量矩阵中词频的稀疏，更多使用TF-IDF值，而不是词频

- 另外也常使用PMI和PPMI值作为向量值计算方法

PMI:

$$\mathrm{PMI}(word_1, word_2) = \log_2 \frac{P(word_1, word_2)}{P(word_1)P(word_2)}$$

PPMI:

将那些PMI值小于0的以0替换。

词-上下文矩阵中PPMI的计算

- 矩阵F，行W为词，列C为上下文
- f_{ij} 表示词w_i出现在上下文c_j中的次数

	aardvark	computer	data	pinch	result	sugar
apricot	0	0	0	1	0	1
pineapple	0	0	0	1	0	1
digital	0	2	1	0	1	0
information	0	1	6	0	4	0

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad p_{i*} = \frac{\sum_{j=1}^{C} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}} \qquad p_{*j} = \frac{\sum_{i=1}^{W} f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

$$pmi_{ij} = \log \frac{p_{ij}}{p_{i*}p_{*j}} \qquad ppmi_{ij} = \begin{cases} pmi_{ij} & \text{if } pmi_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$p_{ij} = \frac{f_{ij}}{\sum_{i=1}^{W}\sum_{j=1}^{C} f_{ij}}$$

Count(w,context)

	computer	data	pinch	result	sugar
apricot	0	0	1	0	1
pineapple	0	0	1	0	1
digital	2	1	0	1	0
information	1	6	0	4	0

p(w=information,c=data) = 6/19 = .32

p(w=information) = 11/19 = .58

p(c=data) = 7/19 = .37

$$p(w_i) = \frac{\sum_{j=1}^{C} f_{ij}}{N} \qquad p(c_j) = \frac{\sum_{i=1}^{W} f_{ij}}{N}$$

p(w,context)

	computer	data	pinch	result	sugar	**p(w)**
apricot	0.00	0.00	0.05	0.00	0.05	0.11
pineapple	0.00	0.00	0.05	0.00	0.05	0.11
digital	0.11	0.05	0.00	0.05	0.00	0.21
information	0.05	0.32	0.00	0.21	0.00	0.58
p(context)	0.16	0.37	0.11	0.26	0.11	

$$pmi_{ij} = \log \frac{p_{ij}}{p_{i*} p_{*j}}$$

	p(w,context)					p(w)
	computer	data	pinch	result	sugar	
apricot	0.00	0.00	0.05	0.00	0.05	0.11
pineapple	0.00	0.00	0.05	0.00	0.05	0.11
digital	0.11	0.05	0.00	0.05	0.00	0.21
information	0.05	0.32	0.00	0.21	0.00	0.58
p(context)	0.16	0.37	0.11	0.26	0.11	

pmi(information,data) = $.32 / (.37*.58)) = \log(1.49)$
$= .39$

PPMI(w,context)

	computer	data	pinch	result	sugar
apricot	-	-	1.56	-	1.56
pineapple	-	-	1.56	-	1.56
digital	1.15	0.00	-	0.00	-
information	0.00	0.39	-	0.32	-

加权PMI

- 加-1平滑
- 加-2平滑
- PPMI

Add-1 Smoothed Count(w,context

	computer	data	pinch	result	sugar
apricot	1	1	2	1	2
pineapple	1	1	2	1	2
digital	3	2	1	2	1
information	2	7	1	5	1

p(w,context) [add-1] **p(w)**

	computer	data	pinch	result	sugar	
apricot	0.03	0.03	0.05	0.03	0.05	0.18
pineapple	0.03	0.03	0.05	0.03	0.05	0.18
digital	0.08	0.05	0.03	0.05	0.03	0.23
information	0.05	0.18	0.03	0.13	0.03	0.41
p(context)	0.18	0.28	0.15	0.23	0.15	

PPMI(w,context)

	computer	data	pinch	result	sugar
apricot	-	-	0.68	-	0.68
pineapple	-	-	0.68	-	0.68
digital	0.50	0.00	-	0.00	-
information	0.00	0.17	-	0.14	-

PPMI(w,context) [add-2]

	computer	data	pinch	result	sugar
apricot	0.00	0.00	0.17	0.00	0.17
pineapple	0.00	0.00	0.17	0.00	0.17
digital	0.19	0.00	0.00	0.00	0.00
information	0.00	0.18	0.00	0.11	0.00

向量之间的相似度计算

余弦相似度

Dot product

Unit vectors

$$\cos(v,w) = \frac{v \bullet w}{|v||w|} = \frac{v}{|v|} \bullet \frac{w}{|w|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}$$

v_i is the PPMI value for word v in context i

w_i is the PPMI value for word w in context i.

Cos($\vec{v}$,$\vec{w}$) is the cosine similarity of $\vec{v}$ and $\vec{w}$

	large	data	computer
apricot	1	0	0
digital	0	1	2
information	1	6	1

$$\cos(v,w) = \frac{v \bullet w}{|v||w|} = \frac{v}{|v|} \bullet \frac{w}{|w|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}$$

Which pair of words is more similar?

cosine(apricot,information) = $\frac{1+0+0}{\sqrt{1+0+0}\sqrt{1+36+1}} = \frac{1}{\sqrt{38}} = .16$

cosine(digital,information) = $\frac{0+6+2}{\sqrt{0+1+4}\sqrt{1+36+1}} = \frac{8}{\sqrt{38}\sqrt{5}} = .58$

cosine(apricot,digital) = $\frac{0+0+0}{\sqrt{1+0+0}\sqrt{0+1+4}} = 0$

向量间相似度度量方法

$$\mathrm{sim}_{\mathrm{cosine}}(\vec{v},\vec{w}) = \frac{\vec{v}\cdot\vec{w}}{|\vec{v}||\vec{w}|} = \frac{\sum_{i=1}^{N} v_i \times w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}}$$

$$\mathrm{sim}_{\mathrm{Jaccard}}(\vec{v},\vec{w}) = \frac{\sum_{i=1}^{N} \min(v_i,w_i)}{\sum_{i=1}^{N} \max(v_i,w_i)}$$

$$\mathrm{sim}_{\mathrm{Dice}}(\vec{v},\vec{w}) = \frac{2\times\sum_{i=1}^{N} \min(v_i,w_i)}{\sum_{i=1}^{N} (v_i+w_i)}$$

$$\mathrm{sim}_{\mathrm{JS}}(\vec{v}||\vec{w}) = D\left(\vec{v}\middle|\frac{\vec{v}+\vec{w}}{2}\right) + D\left(\vec{w}\middle|\frac{\vec{v}+\vec{w}}{2}\right)$$

- 其它最新语义分析技术请参考相关文献。
  - **Efficient Estimation of Word Representations in vector space. Mikolov, 2013.**