《知识图谱: 概念与技术》

第 11 讲
基于知识图谱的语言理解

肖仰华

复旦大学

shawyh@fudan.edu.cn

本章大纲

- 语言理解概述
- 理解实体
- 理解概念
- 其他理解

概述

Why language understanding is so important?

- Language understanding is one of the important thinking activity

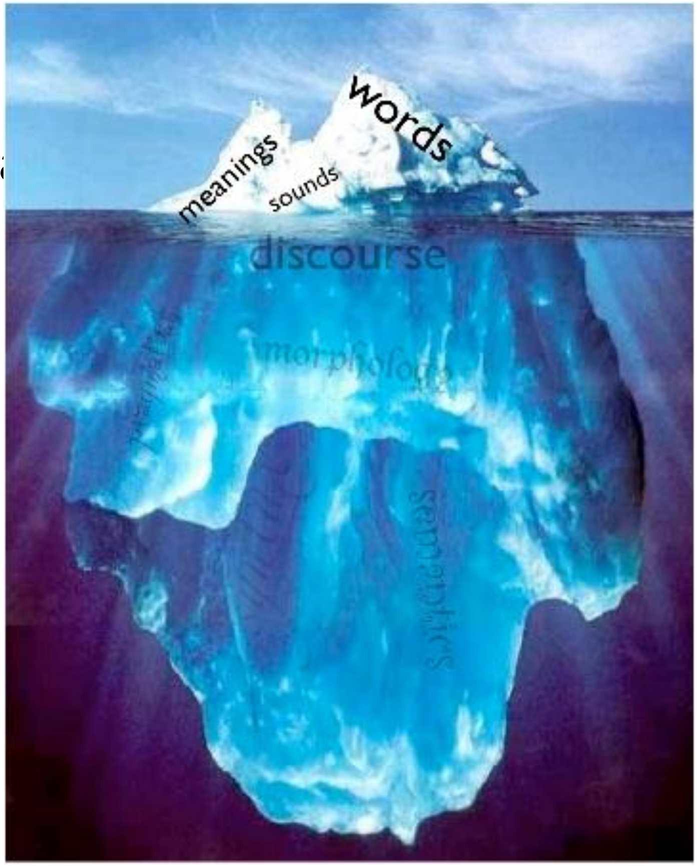
- Language is the tool of thinking
- It is the ability of language speaking and un
  distinguish us from animals

Enabling machine to understand human language is the essential path to realize intelligent information processing and smart robot brain.

Why language understanding is so hard?

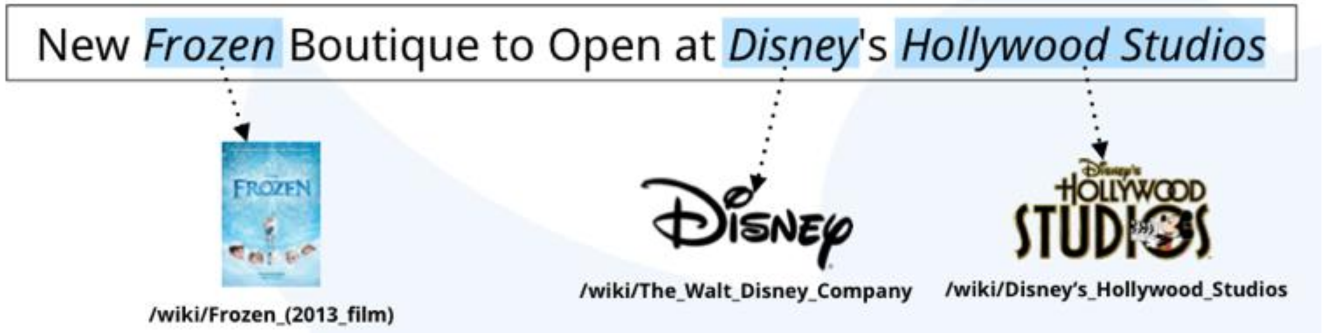
Language is complicated

- **Ambiguous**, **contextual** and **implicit**
- Seemingly **infinite** number of ways to express the same mea



Language understanding is difficult

- Grounded only in **human cognition**
- Needs significant **background knowledge**


New *Frozen* Boutique to Open at *Disney*'s *Hollywood Studios*

FROZEN
/wiki/Frozen_(2013_film)
Disney
/wiki/The_Walt_Disney_Company
Disney's HOLLYWOOD STUDIOS
/wiki/Disney's_Hollywood_Studios

Machine language understanding needs good background knowledge

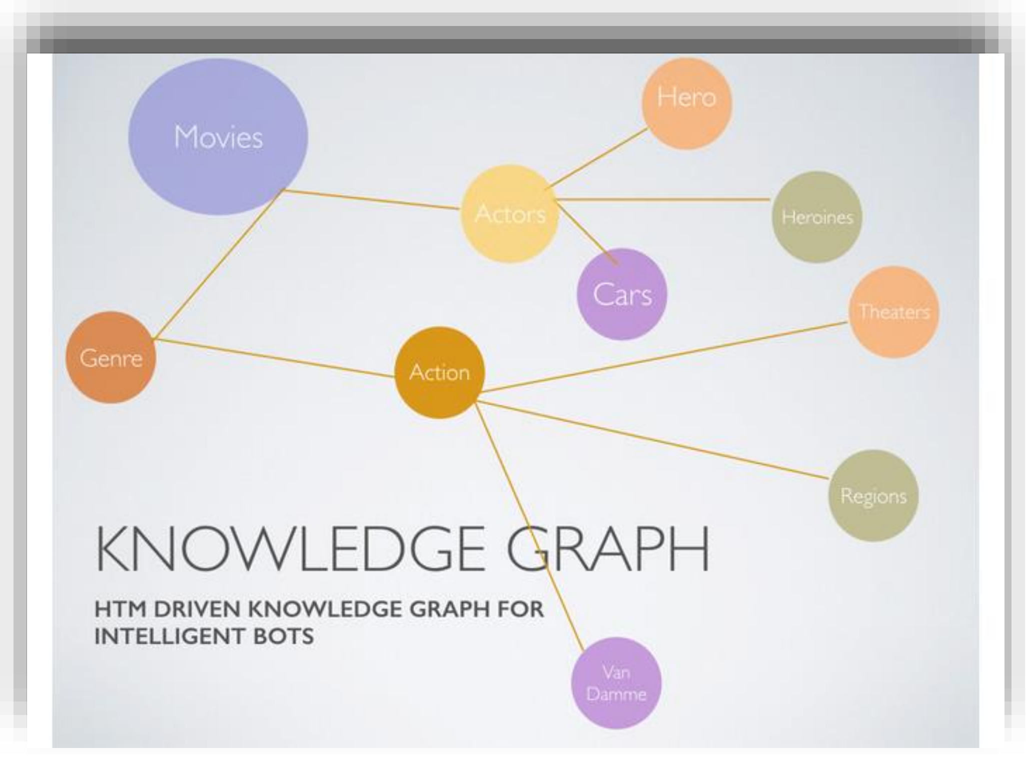
- Language understanding of machines needs knowledge bases
  - Large scale
  - Semantically rich
  - Friendly structure

- Traditional knowledge representations can not satisfy
  - Ontology
  - Semantic network
  - Texts

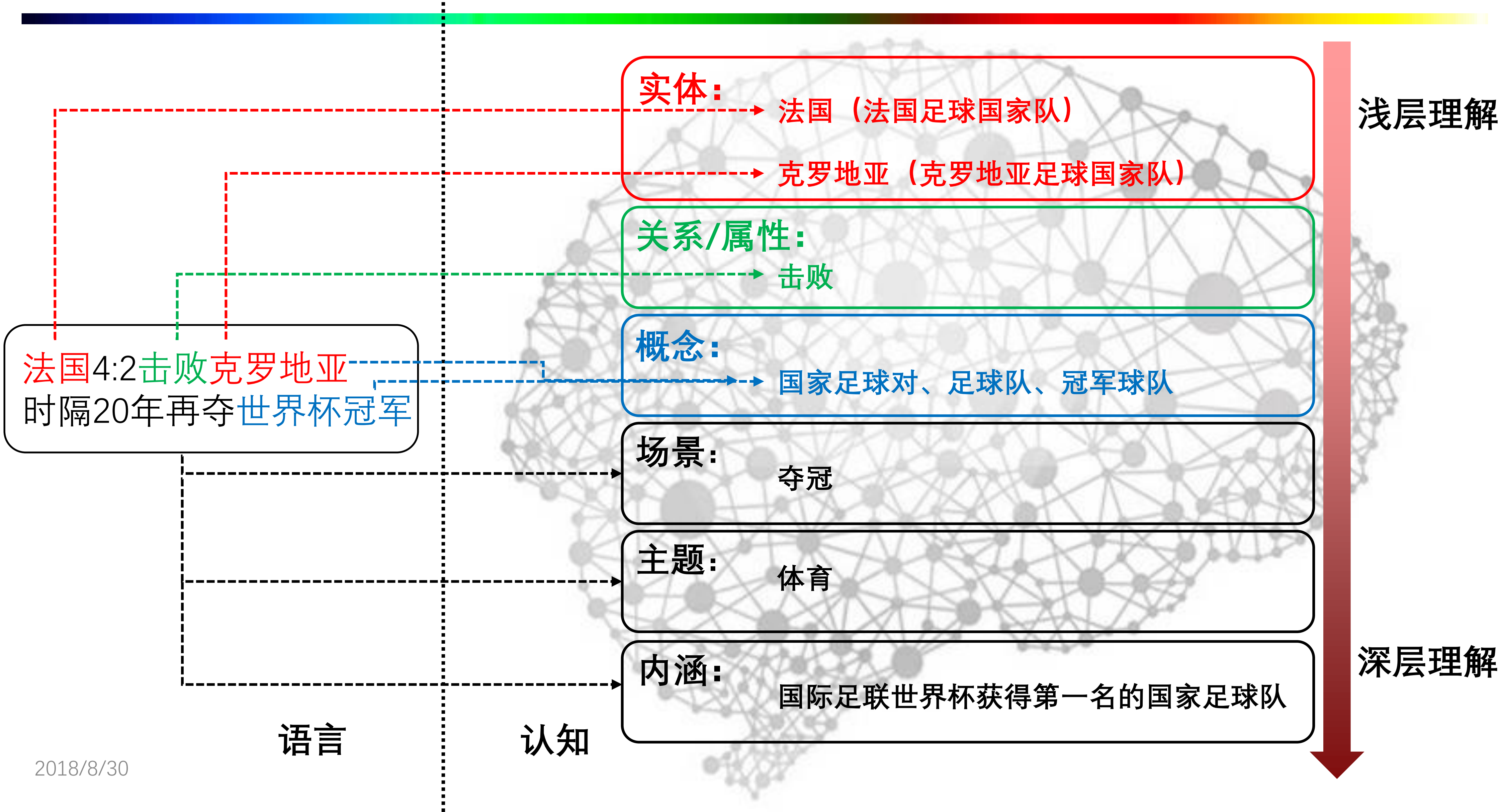
Knowledge graph is a good choice

- Knowledge graph is a large scale semantic network consisting of entities/concepts as well as the semantic relationships among them

- Advantage
  - Higher coverage over entities and concept
  - Richer semantic relationships
  - Usually organized as RDF
  - Quality insurance by Crowdsourcing



语言理解



法国4:2击败克罗地亚
时隔20年再夺世界杯冠军

实体：
法国（法国足球国家队）
克罗地亚（克罗地亚足球国家队）

关系/属性：
击败

概念：
国家足球对、足球队、冠军球队

场景：
夺冠

主题：
体育

内涵：
国际足联世界杯获得第一名的国家足球队

浅层理解

深层理解

语言

认知

实体理解

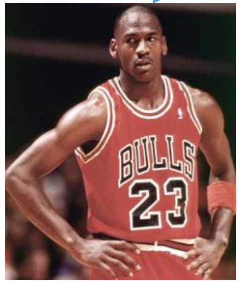
实体理解

- 理解指代：同一个实体存在不同指代（mention）

Barack Obama
Barack H. Obama
President Obama
Senator Obama
President of the United States

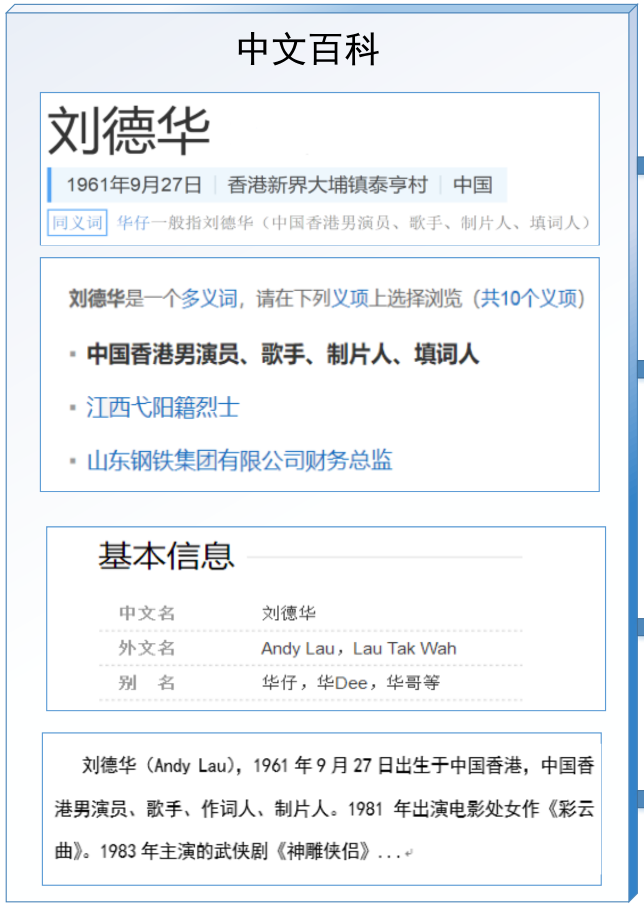
- 实体消歧/实体链接：同一个指代词可能指代多个不同实体

NBA Michael Jordan　　Machine Learning Michael Jordan



理解实体指代

- 利用在线百科丰富的结构化信息构建实体-指代映射表



中文百科

刘德华

1961年9月27日 | 香港新界大埔镇泰亨村 | 中国

同义词 华仔一般指刘德华（中国香港男演员、歌手、制片人、填词人）

刘德华是一个多义词，请在下列义项上选择浏览（共10个义项）

- 中国香港男演员、歌手、制片人、填词人
- 江西弋阳籍烈士
- 山东钢铁集团有限公司财务总监

基本信息

中文名	刘德华
外文名	Andy Lau，Lau Tak Wah
别　名	华仔，华Dee，华哥等

刘德华（Andy Lau），1961 年 9 月 27 日出生于中国香港，中国香港男演员、歌手、作词人、制片人。1981 年出演电影处女作《彩云曲》。1983 年主演的武侠剧《神雕侠侣》...↵

基于同义词的抽取

mention	entity
华仔	刘德华（中国香港男演员、歌手、词作人）

基于多义词的抽取

刘德华	刘德华（中国香港男演员、歌手、词作人）
刘德华	刘德华（山东钢铁集团有限公司财务总监）
刘德华	刘德华（江西弋阳籍烈士）

基于名称类属性的抽取

华仔	刘德华（中国香港男演员、歌手、词作人）
华Dee	刘德华（中国香港男演员、歌手、词作人）
华哥	刘德华（中国香港男演员、歌手、词作人）
andy lau	刘德华（中国香港男演员、歌手、词作人）
lau tak wah	刘德华（中国香港男演员、歌手、词作人）

基于摘要文本的抽取

andy lau	刘德华（中国香港男演员、歌手、词作人）

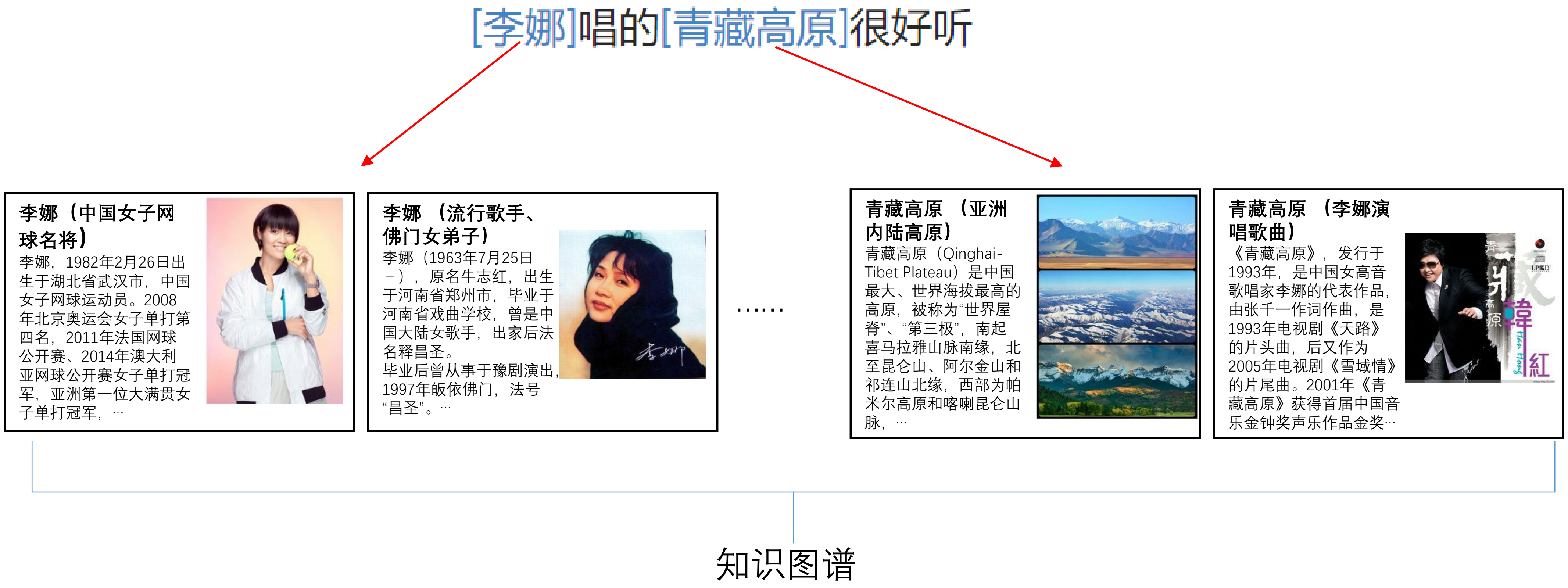
融合

输出

mention	entity
华仔	刘德华（中国香港男演员、歌手、词作人）
刘德华	刘德华（中国香港男演员、歌手、词作人）
刘德华	刘德华（山东钢铁集团有限公司财务总监）
刘德华	刘德华（江西弋阳籍烈士）
华Dee	刘德华（中国香港男演员、歌手、词作人）
华哥	刘德华（中国香港男演员、歌手、词作人）
andy lau	刘德华（中国香港男演员、歌手、词作人）
lau tak wah	刘德华（中国香港男演员、歌手、词作人）

实体链接：问题描述

- 将文本中的实体指代链接到知识库中特定实体
    - 主要问题：指代词歧义



[李娜]唱的[青藏高原]很好听

李娜（中国女子网球名将）
李娜，1982年2月26日出生于湖北省武汉市，中国女子网球运动员。2008年北京奥运会女子单打第四名，2011年法国网球公开赛、2014年澳大利亚网球公开赛女子单打冠军，亚洲第一位大满贯女子单打冠军，…

李娜 （流行歌手、佛门女弟子）
李娜（1963年7月25日－），原名牛志红，出生于河南省郑州市，毕业于河南省戏曲学校，曾是中国大陆女歌手，出家后法名释昌圣。
毕业后曾从事于豫剧演出，1997年皈依佛门，法号“昌圣”。…

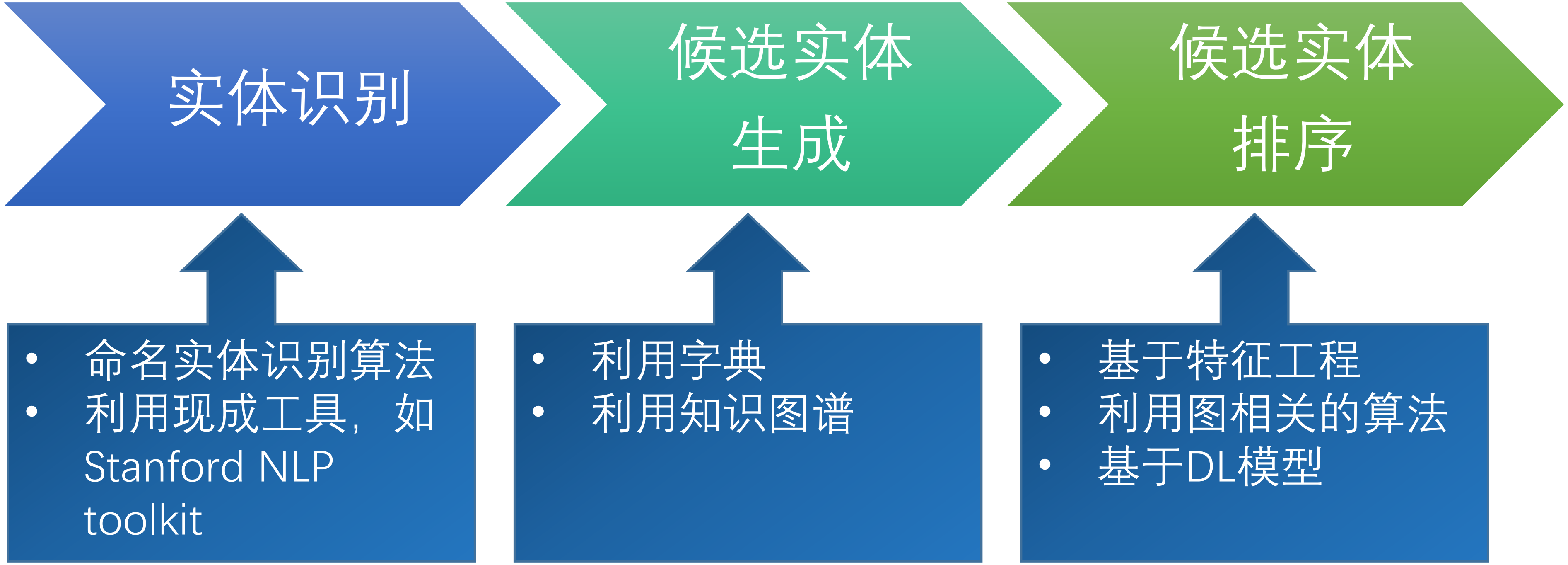
……

青藏高原 （亚洲内陆高原）
青藏高原（Qinghai-Tibet Plateau）是中国最大、世界海拔最高的高原，被称为“世界屋脊”、“第三极”，南起喜马拉雅山脉南缘，北至昆仑山、阿尔金山和祁连山北缘，西部为帕米尔高原和喀喇昆仑山脉，…

青藏高原 （李娜演唱歌曲）
《青藏高原》，发行于1993年，是中国女高音歌唱家李娜的代表作品，由张千一作词作曲，是1993年电视剧《天路》的片头曲，后又作为2005年电视剧《雪域情》的片尾曲。2001年《青藏高原》获得首届中国音乐金钟奖声乐作品金奖…

知识图谱

实体链接：基本流程



实体识别

- 命名实体识别算法
- 利用现成工具，如Stanford NLP toolkit

候选实体生成

- 利用字典
- 利用知识图谱

候选实体排序

- 基于特征工程
- 利用图相关的算法
- 基于DL模型

实体链接：基本模型

- 输入：文本；以及上下文已识别指代集 $M = (m_1, m_2, \dots, m_n)$
- 输出：链接实体列表 $\mathit{\Gamma} = (e_1, e_2, \dots, e_n)$

- 模型：

$$\mathit{\Gamma}_{\text{best}} = \arg max_{\mathit{\Gamma}}\left(\sum_{i=1}^{n}\varphi(m_i, e_i) + \psi(\mathit{\Gamma})\right)$$

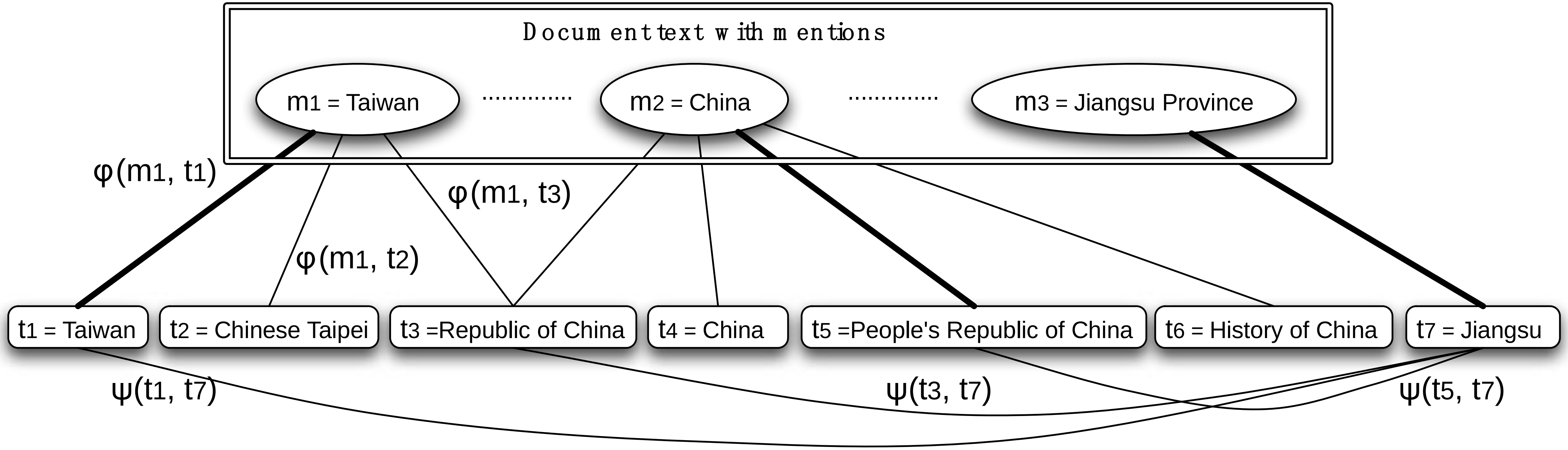
局部匹配函数：刻画实体与上下文的匹配程度计算

$$\varphi(m_i, e_i) = \sum\nolimits_{\mathrm{k}} \mathrm{w_k f_k}(e_i, \mathrm{m_i})$$

全局匹配函数：考虑上下文里实体对之间的语义关联强度

$$\psi(\mathit{\Gamma}) = \sum\nolimits_{e_i \in \mathit{\Gamma}, e_j \in \mathit{\Gamma}} r\,(e_i, e_j)$$

实体链接：模型示例


Document text with mentions
m1 = Taiwan
....
m2 = China
....
m3 = Jiangsu Province
φ(m1, t1)
φ(m1, t3)
φ(m1, t2)
t1 = Taiwan
t2 = Chinese Taipei
t3 =Republic of China
t4 = China
t5 =People's Republic of China
t6 = History of China
t7 = Jiangsu
ψ(t1, t7)
ψ(t3, t7)
ψ(t5, t7)

φ局部特征

- 上下文无关特征
  - 指代与实体名的字符相似度
    - 实体的指代与实体名通常是有一定相似度的
      - Eg：“周杰伦”与“杰伦”、“周董”
  - 实体流行度（popularity）
    - $p(\mathrm{e}=$ 北京(中华人民共和国首都)$) >$
    - $p(\mathrm{e}=$ 北京(北宋时期行政区划)$)>$
    - $p(\mathrm{e}=$ 北京(诗歌)$)$
  - 百科锚文本先验概率
    - $p(e|\mathrm{A}=$ ‘李娜’$)$
    - 从百科文本中统计出来，依赖大量锚文本

- 上下文有关特征
  - 文本相似度：候选实体相关文本与上下文文本的文本相似度
    - Eg，利用候选实体及上下文的词袋向量或概念向量，计算相应的cosine similarity

$$f(m,e)=\frac{\sum_{w\in Context(m)\cup Text(e)}\mathrm{TF}IDF_m(w)\times\mathrm{TF}IDF_e(w)}{\sqrt{\sum_{w\in Context(m)}\mathrm{TF}IDF_m(w)}\times\sqrt{\sum_{w\in Text(e)}\mathrm{TF}IDF_e(w)}}$$

ψ全局特征

- 候选实体(e1)与上下文实体(e2)之间的语义相关度
- 基于实体在知识图谱中的邻居集合U_1和U_2进行评估
  - Jaccard相似度：$JACC(u_1,u_2)=\frac{|U_1 \cap U_2|}{|U_1 \cup U_2|}$
  - 互信息： $PMI(u_1,u_2)=\frac{|U_1 \cap U_2|/|W|}{|U_1|/|\mathrm{W}|*|U_2|/|W|}$
    （联合概率除以各自独立的概率）
  - 规范化谷歌距离： $NGD(u_1,u_2)=1-\frac{\log(\max(|U_1|,|U_2|))-\log(|U_1 \cap U_2|)}{\log(|W|)-\log(\min(|U_1|,|U_2|))}$
    （对集合大小偏差做规范化）
  - Adamic Adar 相似度： $AA(u_1,u_2)=\sum_{n \in A \cap B}\log\left(\frac{1}{degree(n)}\right)$
    （对popular的公共邻居进行惩罚）

复杂度

- 最优实体链接问题是NP难问题
- 链接方案Γ的搜索空间为$O(N^2 \times M^N)$
- 近似算法
  - 局部化
  - 图算法

$$\Gamma_{best} = argmax_{\Gamma} \underbrace{\boxed{\sum_{i=1}^{N} \Bigg(\varphi(m_i, e_i)}}_{O(M^N)} + \underbrace{\boxed{\sum_{e_j \in \Gamma} r(e_i, e_j)}}_{O(N^2)} \Bigg)$$

局部近似计算

- 原目标函数

$$\Gamma_{best} = argmax_{\Gamma} \sum_{i=1}^{N} \left(\varphi(m_i, e_i) + \sum_{e_j \in \Gamma} r\left(e_i, e_j\right) \right)$$

- 先只利用局部函数φ算出一个局部最优方案

$$\Gamma_{\mathrm{local}} = argmax_{\Gamma} \sum_{\mathrm{i}=1}^{N} \varphi(m_i, e_i)$$

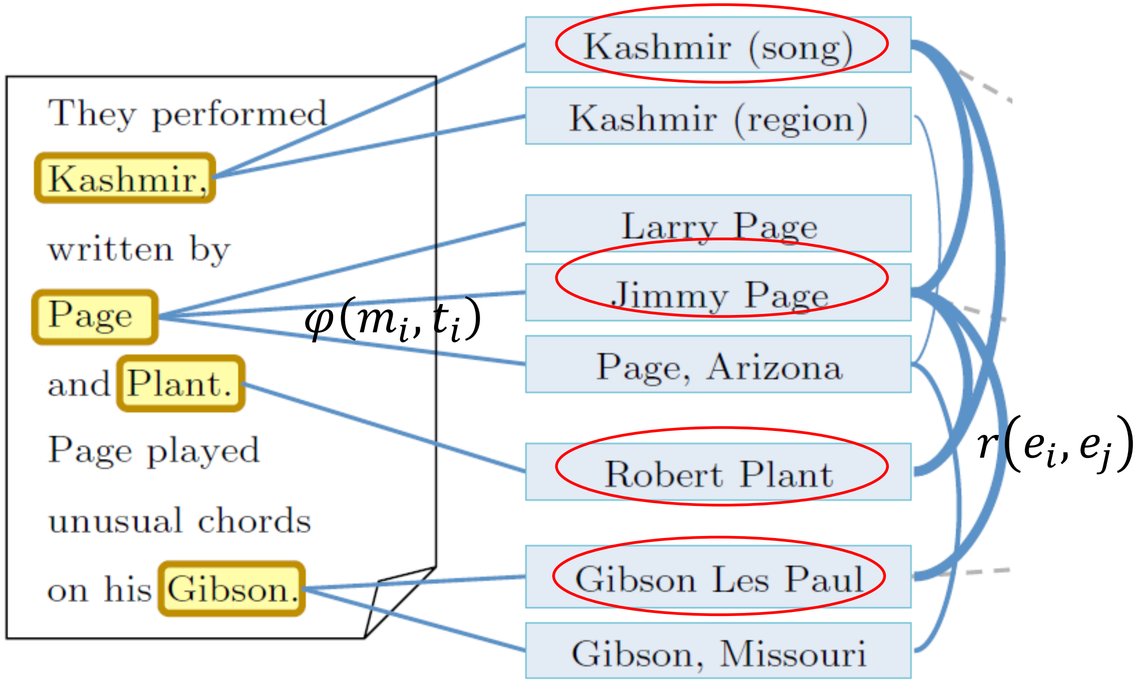
- 然后新的目标函数

$$\Gamma_{best} \approx argmax_{\Gamma} \sum_{i=1}^{N} \left(\varphi(m_i, e_i) + \sum_{e_j \in \Gamma_{\mathrm{local}}} r\left(e_i, e_j\right) \right)$$

- 复杂度：$O(N^2 \times M)$

图算法

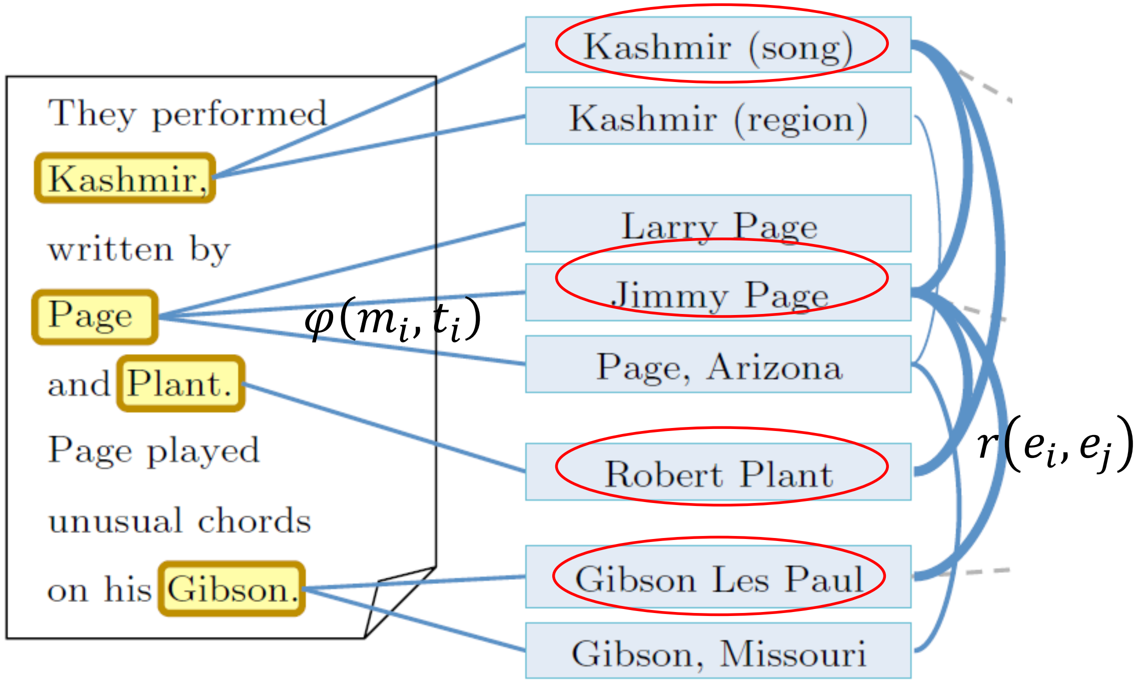
- Mention-Entity Graph
  - 点：指代与实体
  - 边
    - 指代-实体边权：局部分数$\varphi(m_i, e_i)$
    - 实体-实体边权：实体相关度$r(e_i, e_j)$
- 问题模型
  - Given a mention-entity graph, find a *subgraph* with **maximal accumulative weight**
    - Such that contain **all mention nodes** and **exactly one mention-entity edge for each mention**



- NP难(Steiner-tree problem)：贪心算法

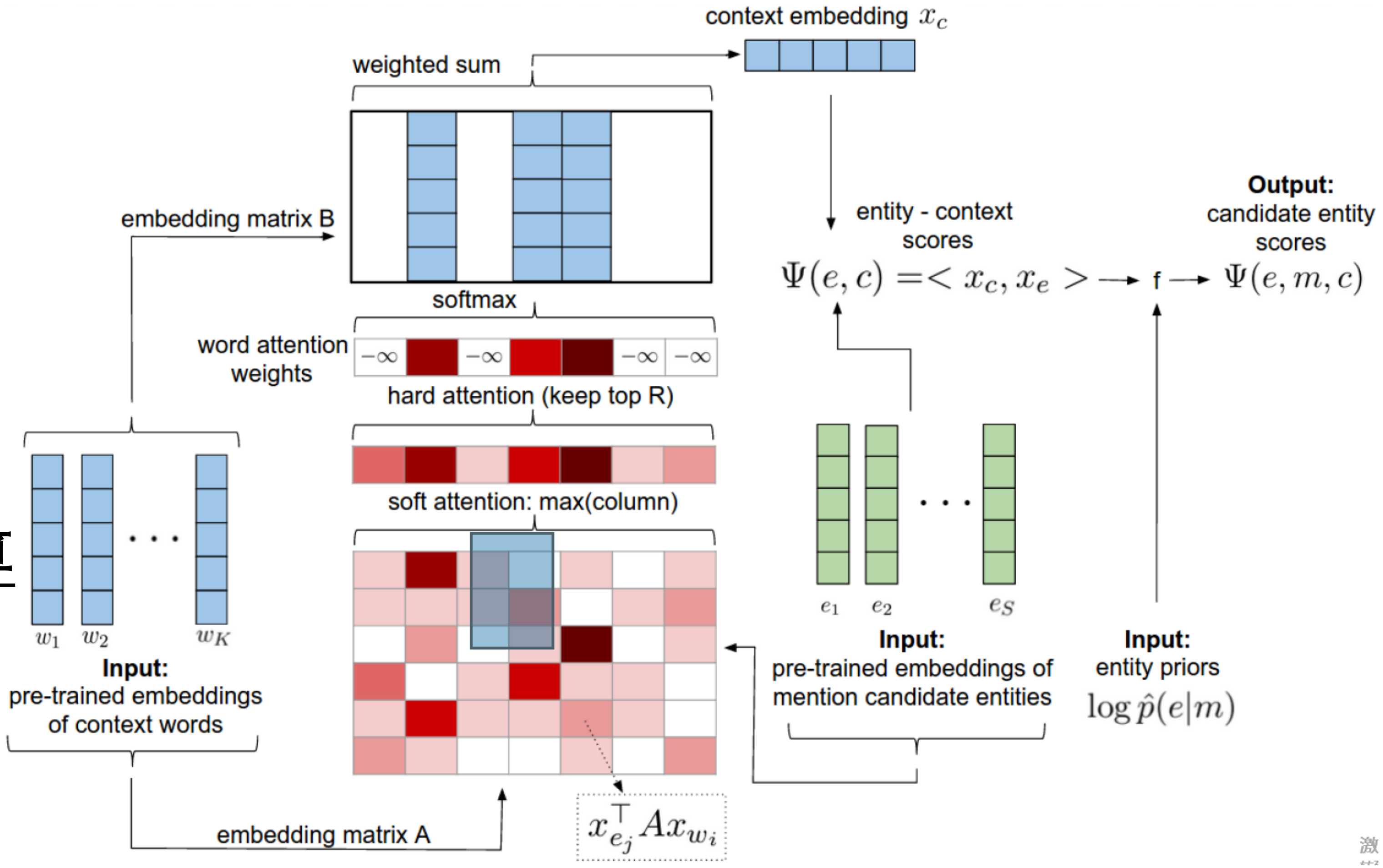
图修剪算法

- 输入：实体与指代的加权图
- 输出：每个指代只有一条边的子图
- 算法流程
    - 对于每个实体，计算它们距离所有指代节点的距离
    - 保留最近的(5×指代数量)个实体, 舍弃其他的
    - 当图中存在非唯一实体时
        - 删除掉加权度最小的非唯一实体以及相邻边
        - 如果删掉的是加权度最小的点，则将当前剩下的图加入解的集合
    - 在最后解的集合里暴力枚举搜索最优的方案



深度实体链接模型

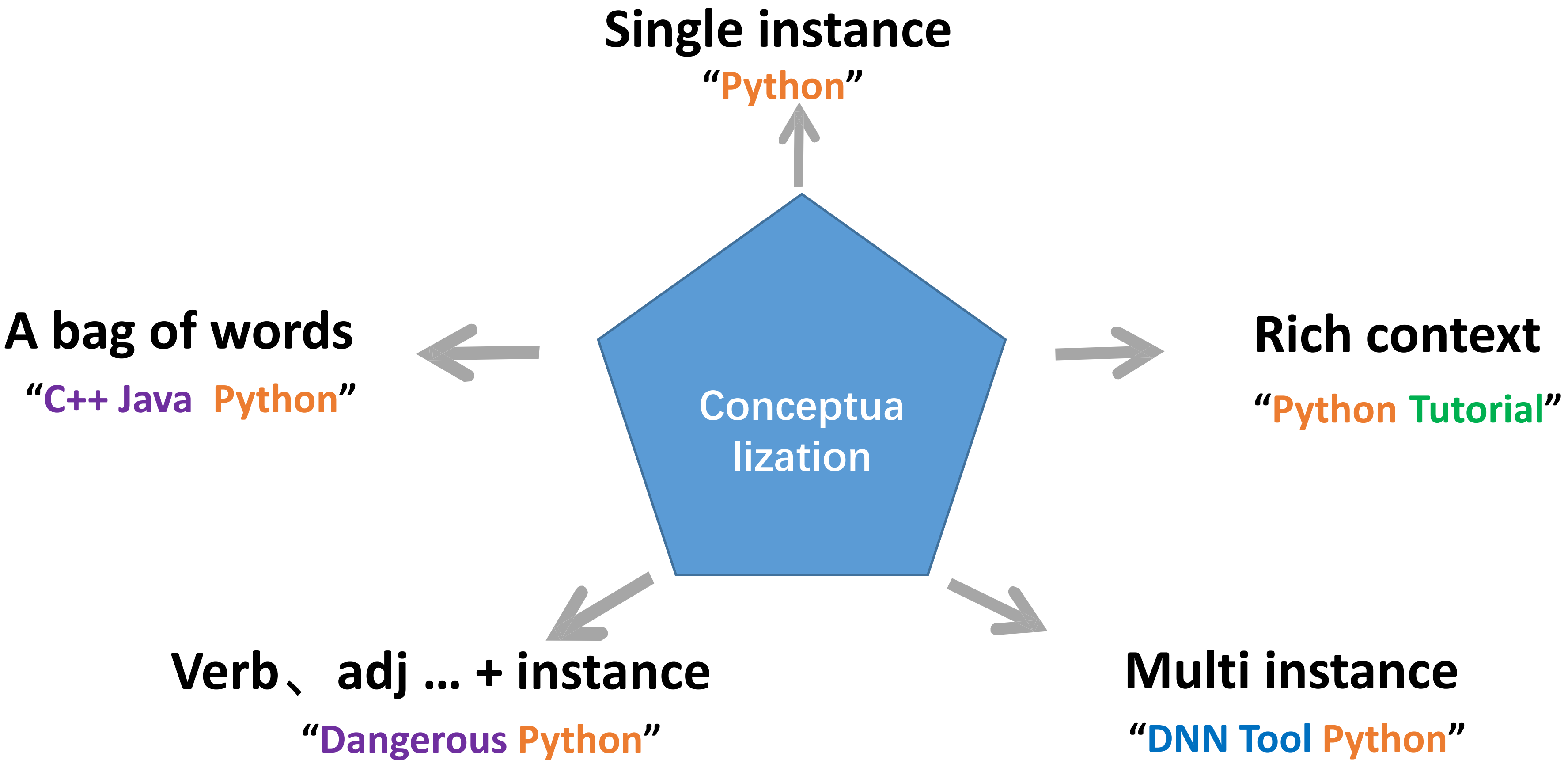
- 无需人工构造特征
- 能够捕捉隐式特征
- 与传统模型优化目标相同
  - $o(\Gamma) = \sum_{i=1}^{n} \varphi(m_i, e_i) + \psi(\Gamma)$
  - ***φ与ψ均由深度方法计算***


context embedding x_c
weighted sum
embedding matrix B
softmax
word attention weights
$-\infty$ $-\infty$ $-\infty$ $-\infty$
hard attention (keep top R)
soft attention: max(column)
entity - context scores
$\Psi(e,c) =< x_c, x_e > \rightarrow$ f $\rightarrow \Psi(e,m,c)$
Output:
candidate entity scores
e_1 e_2 . . . e_S
Input:
pre-trained embeddings of mention candidate entities
Input:
entity priors
$\log \hat{p}(e|m)$
w_1 w_2 . . . w_K
Input:
pre-trained embeddings of context words
embedding matrix A
$x_{e_j}^{\top} A x_{w_i}$

利用注意力机制，上下文中只有少部分词对消歧有影响

概念理解

概念理解Understanding



Conceptualization

Single instance

“Python”

A bag of words

“C++ Java Python”

Rich context

“Python Tutorial”

Verb、adj … + instance

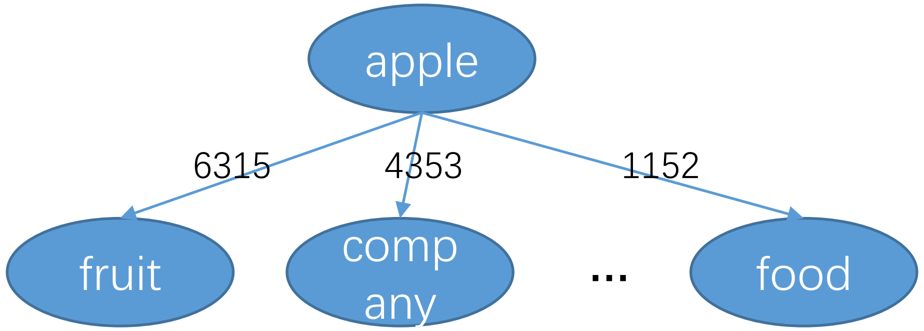
“Dangerous Python”

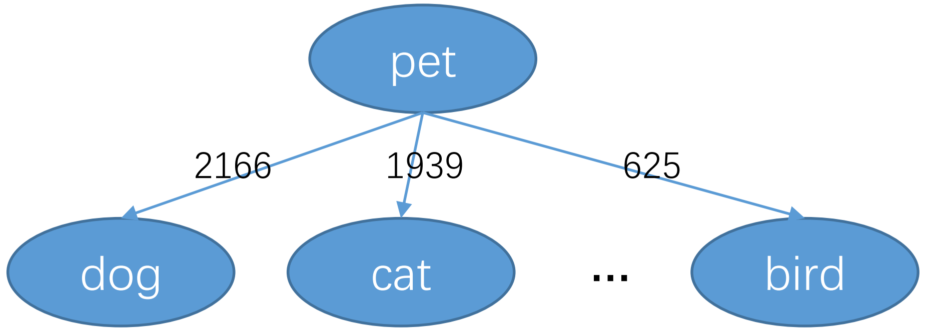
Multi instance

“DNN Tool Python”

Single Instance

- Each isA relation is associated with a frequency
  - that the isA pair is observed in corpus
- Typicality
  - How likely a real human will think up with c (or e) given e (or c)

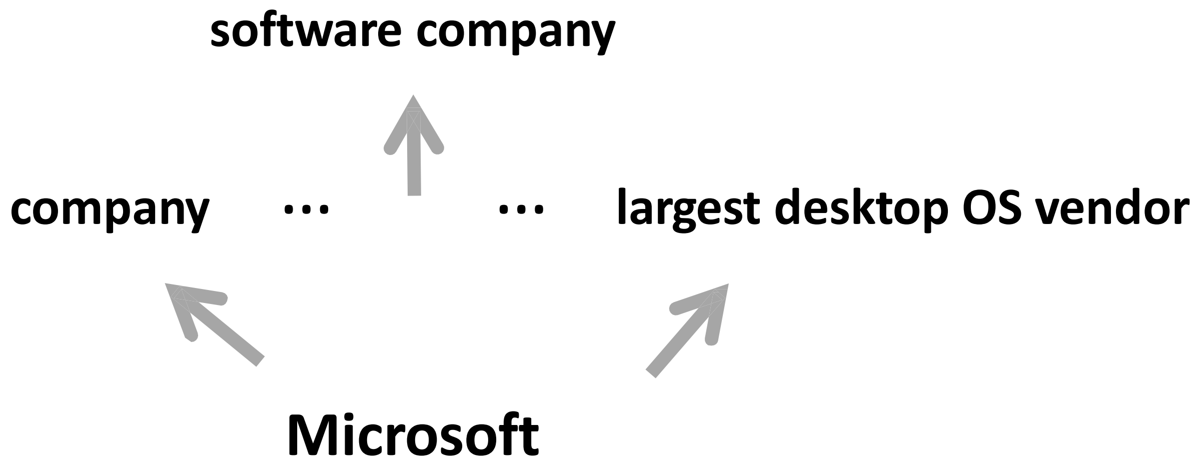




- $p(c|e) = \frac{n(e,c)}{\sum_{c_i} n(e,c_i)}$
- $p(fruit|apple) = \frac{n(apple,fruit)}{n(apple,fruit)+\cdots+n(apple,food)}$

- $p(e|c) = \frac{n(e,c)}{\sum_{e_i} n(e_i,c)}$
- $p(dog|pet) = \frac{n(dog,pet)}{n(dog,pet)+\cdots+n(bird,pet)}$

Single Instance- Basic level categorization



software company

company … … largest desktop OS vendor

Microsoft

Category Level	Informative?	Distinctive?
Superordinate	No	Yes
Basic-level	Yes	Yes
Subordinate	Yes	No

Basic-level conceptualization

Model:

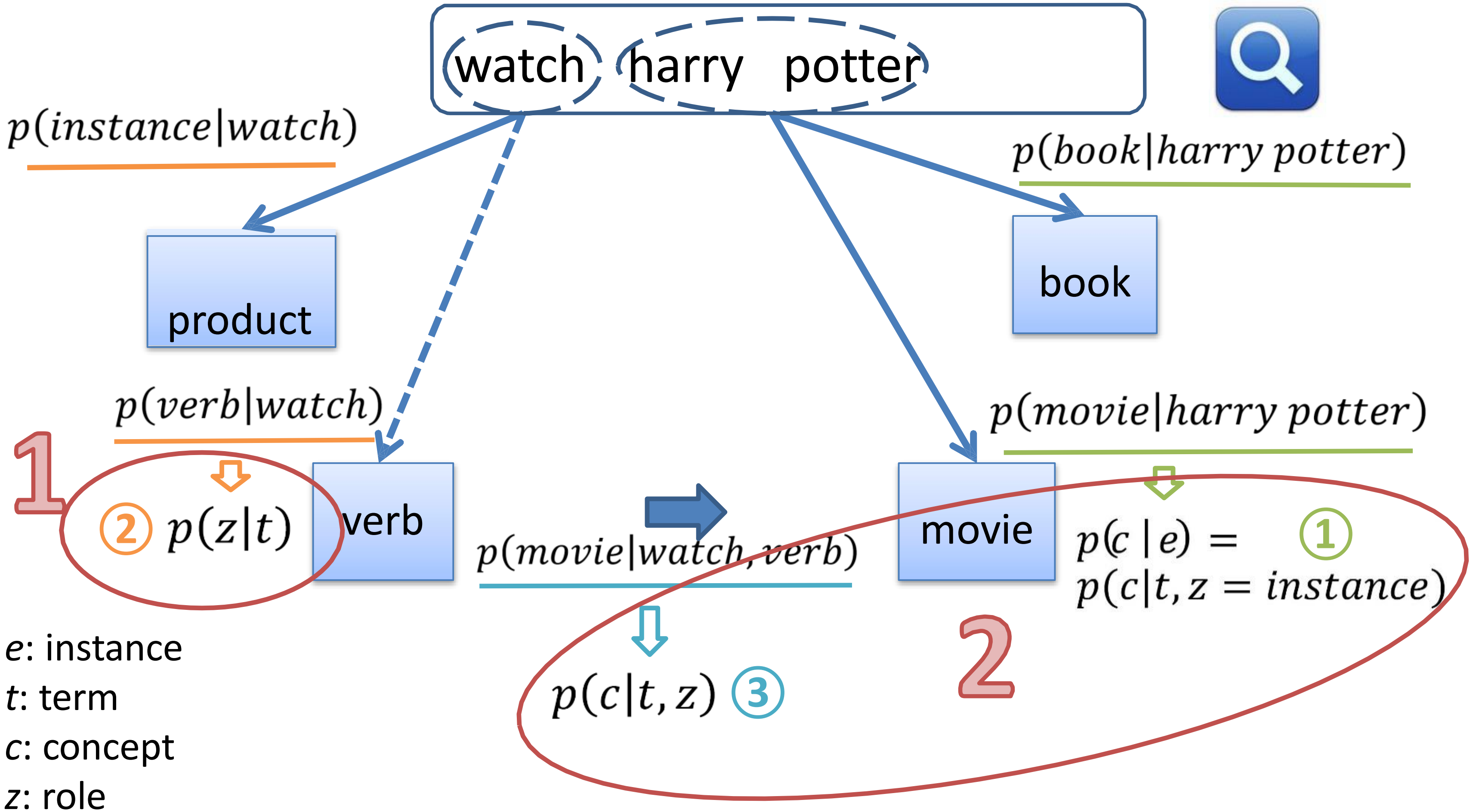
$$Rep(e,c) = P(c|e) * P(e|c)$$

A process of finding *concept nodes* having shortest expected distance with *e*

Given *e*, the *c* should be its typical concept (shortest distance)

Given *c*, the *e* should be its typical entity (shortest distance)

Verb、adj … + instance



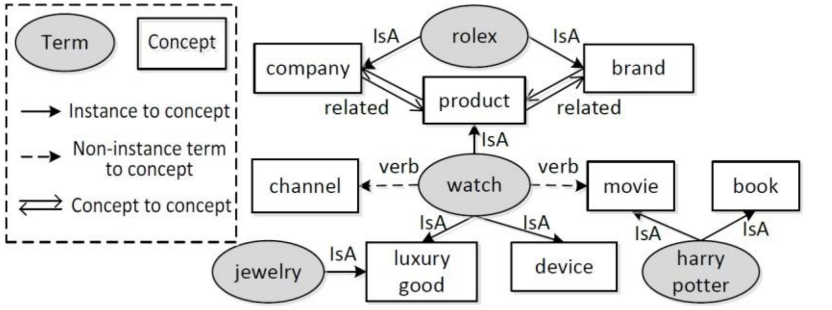
e: instance
t: term
c: concept
z: role

Verb、 adj … + instance

- Understanding Queries: to rank the concepts and find

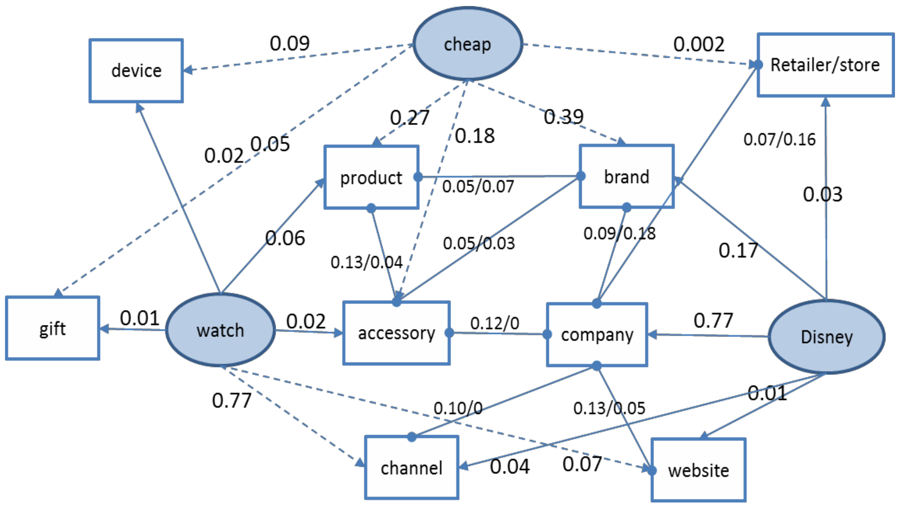
$$\arg\max_{c} p(c|t,q)$$

Query ⇨ Segmentation



The offline semantic network

⇨



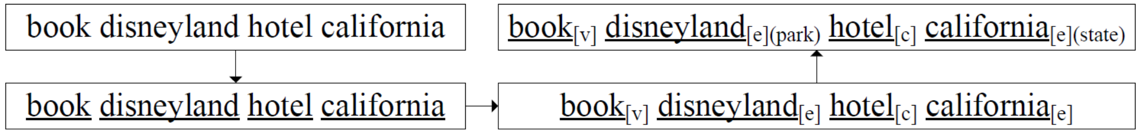
Random walk with restart [Sun et al., 2005] on the online subgraph

Rich context

Problem:

How do we understand short text?

Example:



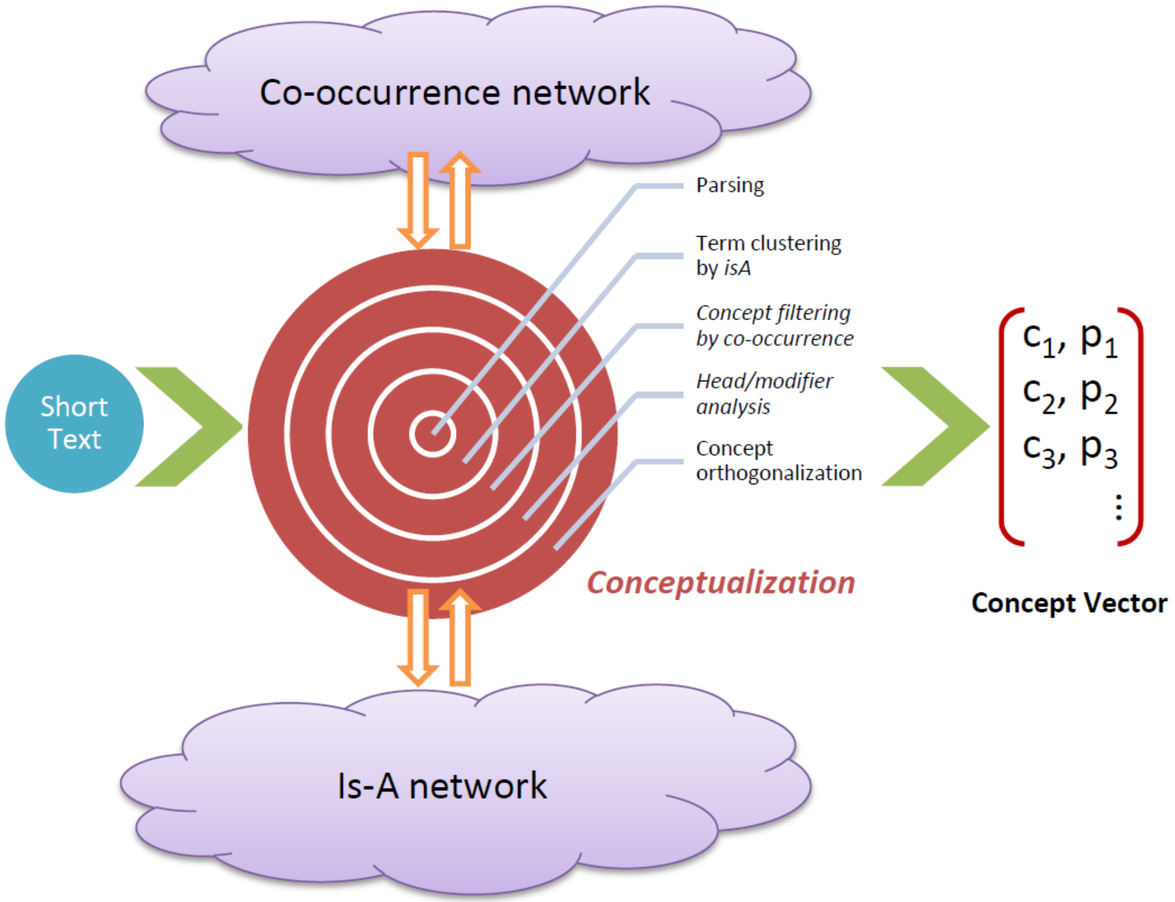
book disneyland hotel california

book disneyland hotel california

book$_{[v]}$ disneyland$_{[e](park)}$ hotel$_{[c]}$ california$_{[e](state)}$

book$_{[v]}$ disneyland$_{[e]}$ hotel$_{[c]}$ california$_{[e]}$

- Text Segmentation
- Type Detection
- Instance Disambiguation



Co-occurrence network

Short Text

Parsing

Term clustering by *isA*

Concept filtering by co-occurrence

Head/modifier analysis

Concept orthogonalization

Conceptualization

c_1, p_1

c_2, p_2

c_3, p_3

⋮

Concept Vector

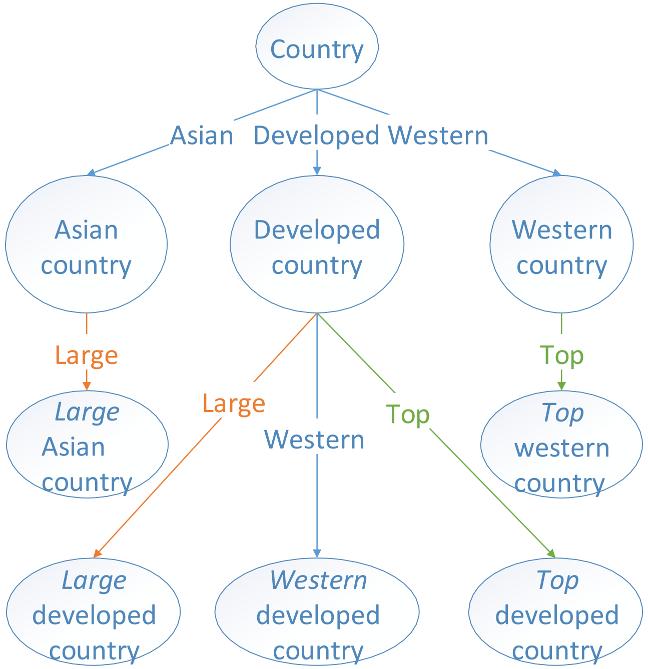
Is-A network

Multi instance

Problem:

Head, Modifier, and Constraint Detection in Short Texts

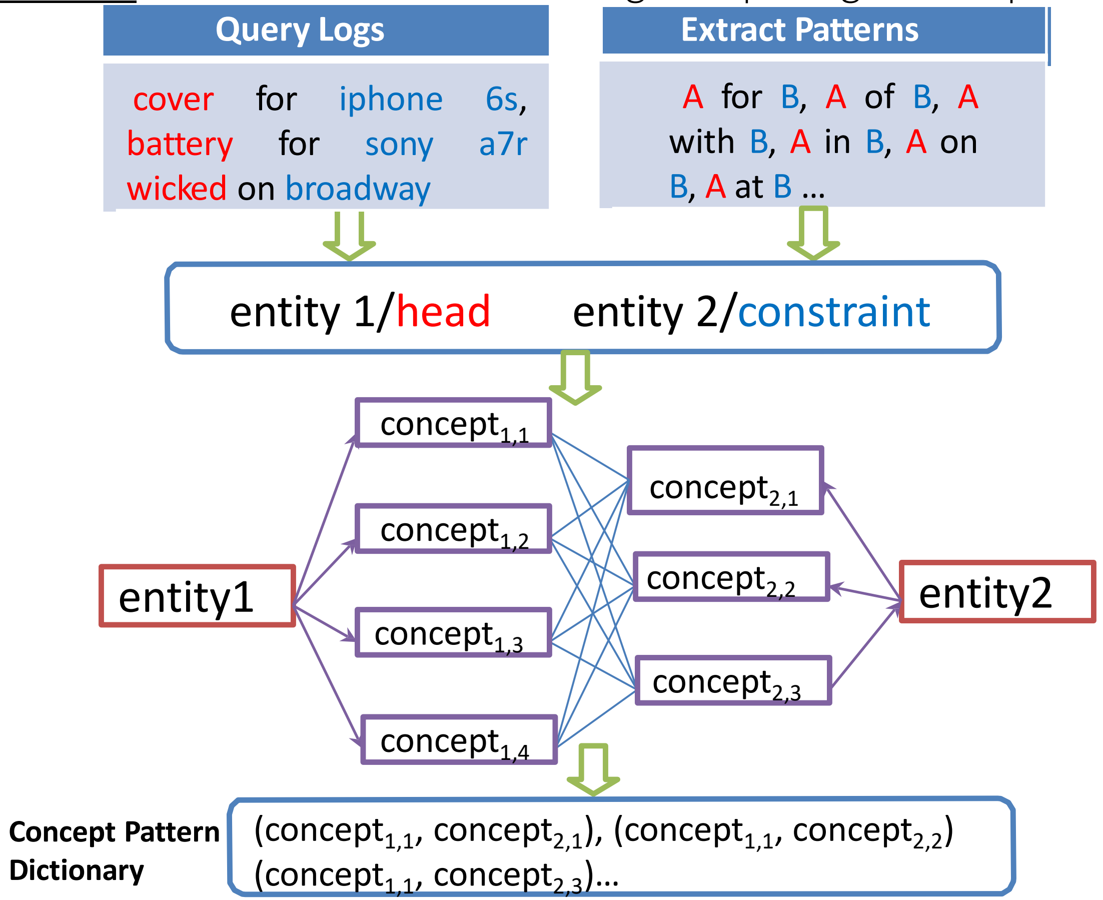
Model1: Non-Constraint Modifiers Mining: Construct Modifier Networks



“Large” and “Top” are pure modifiers

Example: Popular$_{\text{[modifier]}}$ smart cover$_{\text{[head]}}$ iphone 5s$_{\text{[constraint]}}$

Model2: Head-Constraints Mining: Acquiring Concept Patterns



Query Logs

cover for iphone 6s, battery for sony a7r wicked on broadway

Extract Patterns

A for B, A of B, A with B, A in B, A on B, A at B ...

entity 1/head entity 2/constraint

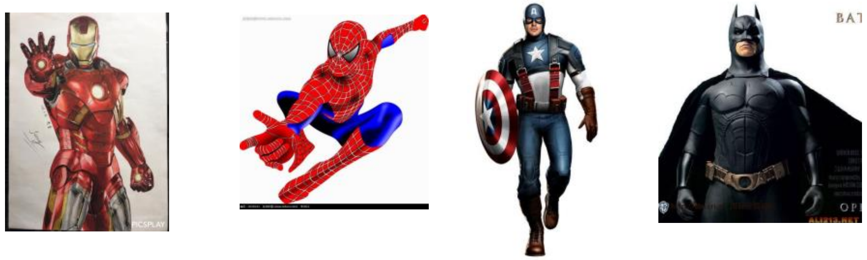
Concept Pattern Dictionary

(concept$_{1,1}$, concept$_{2,1}$), (concept$_{1,1}$, concept$_{2,2}$) (concept$_{1,1}$, concept$_{2,3}$)...

Understanding a bag of words

- Humans understand a bag of words (a set of entities) at the appropriate concept level
- Problems: find a set of concepts for a bag of words (or a set of entities)

Input		**Output**
China, Japan, India, Korea	→	**Asian country**
dinner, lunch, food, child, girl	→	**meal, child**
bride, groom, dress, celebration	→.	**wedding**

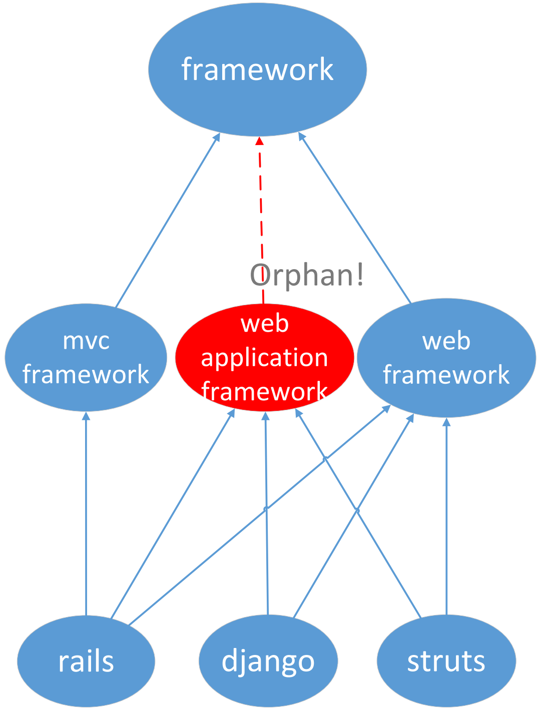


→ **Cartoon Hero**

Understanding a bag of words

- Challenge
  - **Coverage**: cover as many as possible the tags
  - **Minimality**: use the most specific concepts
- Idea
  - Using the isA taxonomy Probase to get the candidate concepts
  - Using minimal description length to select the best concepts

- **Applications**
  - **Topic labelling**
    - A topic is a bag of words that do not have explicit semantics
    - Conceptual labeling turns each topic into a small set of meaningful concepts
  - **Language understanding**
    - Verb role labeling，summarize verb *eat*'s direct objects *apple*, *breakfast*, *pork*, *beef*, into a small set of concepts, such as *fruit*, *meal*, *meat*, *bullet*



Probase

MDL based conceptual labelling

- Problem Model: Given a bag of tags X, find

$$C^* = \operatorname{argmin} CL(X, C)$$

$$CL(X,C) = \underline{L(C)} + \underline{L(X|C)}$$

Minimality: The description length to code concepts

Coverage: The description length to code words by concepts

Improvement

- Noise tolerant
  - Example: {apple, banana, breakfast, dinner, pork, beef, **bullet**}

$$L^*(x|C) = \min \begin{cases} L(x), & \text{encode directly} \\ \log|C| + L(x|c), & \text{encode using } c \in C \end{cases}$$

- Integrating attribute information
  - Example: {population, president, location}, which triggers the concept country

$$P(c|x) = 1 - (1 - P_e(c|x))(1 - P_a(c|x))$$

其他

Understanding a Concept/Category

What is a **bachelor**?

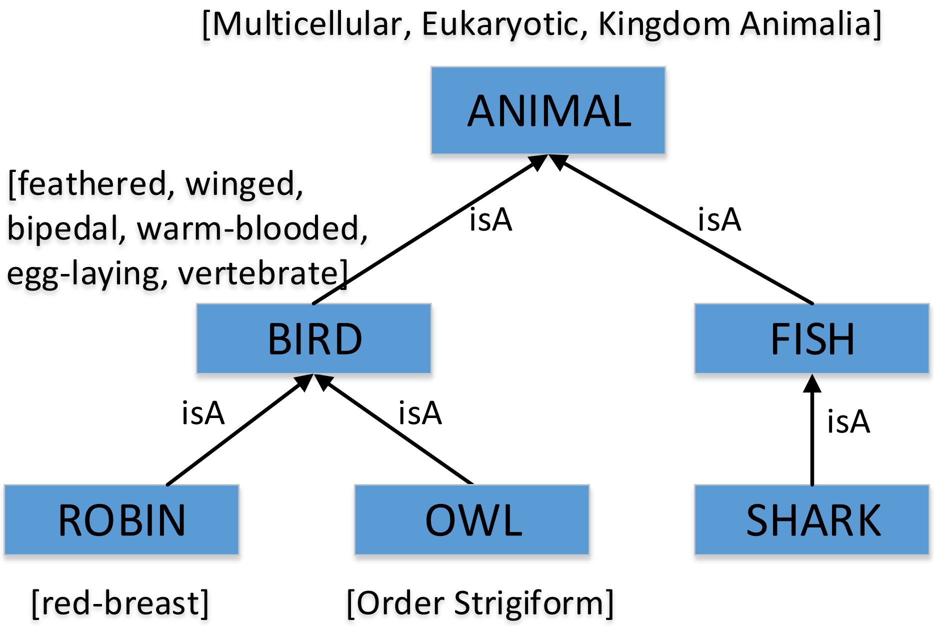
A man that is unmarried

{Type=human, Sex=man, Marriage status=unmarried}

Problem Statement

- Input: a category
- Output: a set of defining features
    - **A most specific type** and a set of Property-Value pairs.
        - E.g., Category "*Jay Chou albums*" is defined by type ***album***, and PV feature ***(Singer, Jay Chou).***

- Defining features are assumed to establish the necessary and sufficient conditions to characterize the meaning of the category.

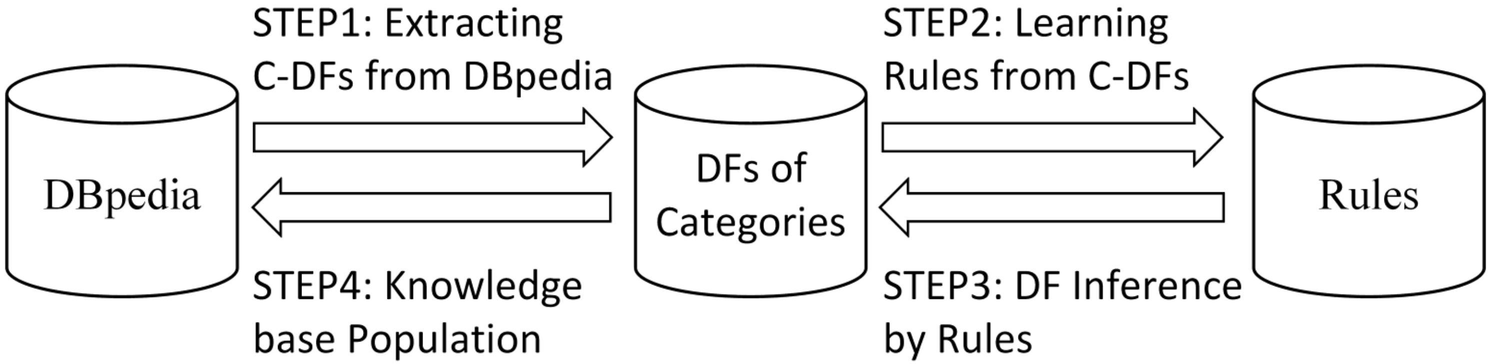


[Multicellular, Eukaryotic, Kingdom Animalia]
ANIMAL
[feathered, winged, bipedal, warm-blooded, egg-laying, vertebrate]
isA
isA
BIRD
FISH
isA
isA
isA
ROBIN
OWL
SHARK
[red-breast]
[Order Strigiform]

Applications: build Semantic memory for machines

Defining Feature Mining

- Framework



DBpedia
STEP1: Extracting C-DFs from DBpedia
DFs of Categories
STEP2: Learning Rules from C-DFs
Rules
STEP3: DF Inference by Rules
STEP4: Knowledge base Population

- A bootstrapping approach
  - Step 1: Using a score function to find DFs of some categories
  - Step 2 & 3: Using a rule-based method to get more DFs of categories
  - Step 4: Populate DBpedia by using DFs of categories discovered so far

Problem Model

- Problem model

$$\hat{\mathbf{f}}(c) = \arg\max_{\mathbf{f}} score(c, \mathbf{f})$$

- The "goodness" of a feature set ***f*** to be the DFs for a category ***c*** is defined as following:

$$score(c, \mathbf{f}) = P(\mathbf{f}|c) \times P(c|\mathbf{f})$$

$$P(\mathbf{f}|c) = \frac{\#\ \textit{of entities in c that have}\ \mathbf{f}}{\#\ \textit{of entities in c}}$$

$$P(c|\mathbf{f}) = \frac{\#\ \textit{of entities in c that have}\ \mathbf{f}}{\#\ \textit{of entities that have}\ \mathbf{f}}$$

- Challenge
  - Naïve solution needs to compute the 'goodness' score for each possible feature sets ($2^N - 1$), where N is the number of candidate features
  - ***Frequent itemset mining*** is used for pruning

Inference Rule of Defining Features

C	DFs
Films directed by Christopher Nolan	{(Type, Film), (director, Christopher Nolan)}
Films directed by James Cameron	{(Type, Film), (director, James Cameron)}
Films directed by Steven Spielberg	{(Type, Film), (director, Steven Spielberg)}
Films directed by David Fincher	{(Type, Film), (director, David Fincher)}
Films directed by Ben Affleck	{(Type, Film), (director, Ben Affleck)}

C Pattern	DFs Pattern
Films directed by (.*)	{(Type, Film), (director, (.*))}

rdfs:range Person

Left part of rule	Right part of rule
Films directed by <Person>	{(Type, Film), (director, <Person>)}

Understanding Verb Phrase

E.g. *I watched The Amazing Spider-man 2 and thought it's impressive.*

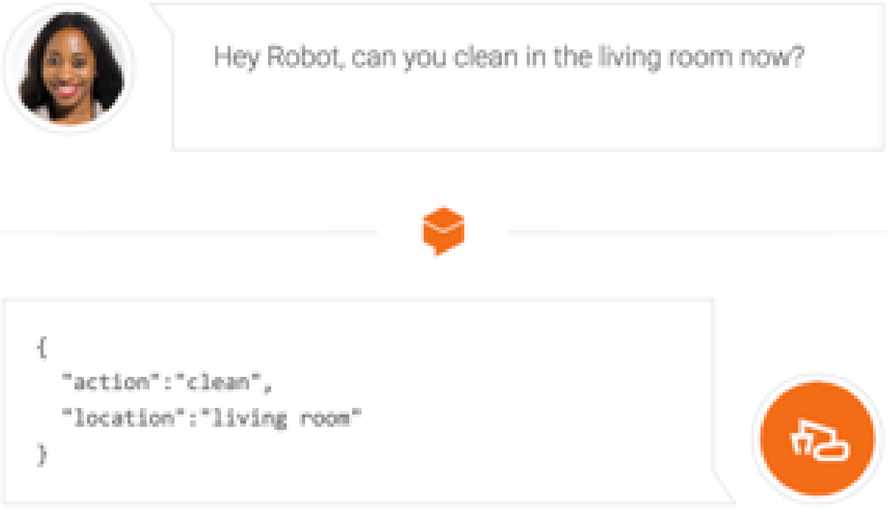
How to understand "The Amazing Spider-man 2" using verb "watch"?

Pattern: watch $movie -> "The Amazing Spider-man 2" isA movie

Linguists [Sinclair 1990] found two principles for verb phrases:

- **idiom patterns**: Kick the ass/ watch step
- **conceptualized patterns**: eat fruit (apple/ banana etc.) drink beverage (wine, tea etc.)

Model: extract the patterns of verb phrases



Hey Robot, can you clean in the living room now?

```
{
  "action":"clean",
  "location":"living room"
}
```

API.AI

Applications:

Conceptualization using verbs

The apple(object) he ate(verb) yesterday has a bad taste.

Pattern: eat $food -> apple isA $food

Parsing: Finding subjective/objective/etc. of a verb

Understanding Verb Phrases

Challenge: trade-off between generality and specificity

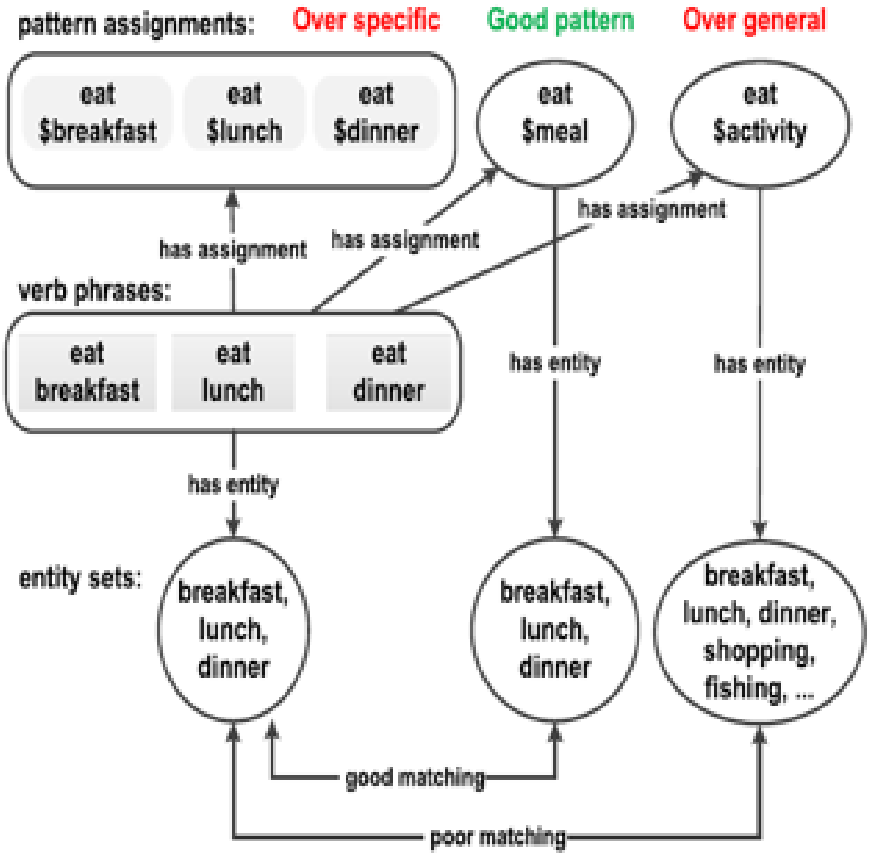
Generality: One general pattern is better than several specific pattern.

Specificity: A pattern's assigned entities and the pattern itself should be matched.

By using MDL:

$$\arg\min_f L(f)$$

$$L(f) = \sum_p P(p)L(p) = \sum_p P(p)[l(p,f) + r(p,f)]$$



References

- Ratinov, Lev-Arie et al. “Local and Global Algorithms for Disambiguation to Wikipedia.” ACL (2011).
- Hoffart, Johannes et al. “Robust Disambiguation of Named Entities in Text.” *EMNLP*(2011).
- Ganea, Octavian-Eugen and Thomas Hofmann. “Deep Joint Entity Disambiguation with Local Neural Attention.” *EMNLP* (2017).
- Bo Xu, Chenhao Xie, Yi Zhang, **Yanghua Xiao***, Haixun Wang and Wei Wang, Learning Defining Features for Categories, (***IJCAI 2016***)
- Zhongyuan Wang, Haixun Wang, Jirong Wen and **Yanghua Xiao**, An Inference Approach to Basic Level of Categorization, (***CIKM 2015***)
- Xiangyan Sun, **Yanghua Xiao***, Haixun Wang, Wei Wang, On Conceptual Labeling of a Bag of Words, (***IJCAI 2015***)
- Wanyun Cui, Xiyou Zhou, Hangyu Lin,**Yanghua Xiao***,Seungwon Hwang,Haixun Wang and Wei Wang, Verb Pattern: A Probabilistic Semantic Representation on Verbs, (***AAAI 2016***)
- Zhongyuan Wang and Haixun Wang, Understanding Short Texts, in *the Association for Computational Linguistics (ACL) (Tutorial)*, August 2016.
- Wen Hua, Zhongyuan Wang, Haixun Wang, Kai Zheng, and Xiaofang Zhou, Understand Short Texts by Harvesting and Analyzing Semantic Knowledge, in *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Volume: PP, Issue: 99, May 23, 2016.
- Zhiyi Luo, Yuchen Sha, Kenny Zhu, Seung-Won Hwang, and Zhongyuan Wang, Commonsense Causal Reasoning between Short Texts, in *the 15th International Conference on Principles of Knowledge Representation and Reasoning (KR 2016)*, April 2016.
- Peipei Li, Haixun Wang, Kenny Q Zhu, Zhongyuan Wang, Xuegang Hu, and Xindong Wu, A Large Probabilistic Semantic Network Based Approach to Compute Term Similarity, in *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Volume: 27, Issue: 10, October 1 2015.

References

- Zhongyuan Wang, Haixun Wang, Ji-Rong Wen, and Yanghua Xiao, An Inference Approach to Basic Level of Categorization, in *ACM International Conference on Information and Knowledge Management (CIKM)*, ACM – Association for Computing Machinery, October 2015.
- Jianpeng Cheng, Zhongyuan Wang, Ji-Rong Wen, Jun Yan, and Zheng Chen, Contextual Text Understanding in Distributional Semantic Space, in *ACM International Conference on Information and Knowledge Management (CIKM)*, ACM – Association for Computing Machinery, October 2015.
- Zhongyuan Wang, Fang Wang, Ji-Rong Wen, and Zhoujun Li, Bring User Interest to Related Entity Recommendation, in *the 4th IJCAI International Workshop on Graph Structures for Knowledge Representation and Reasoning (GKR 2015)*, July 2015.
- Zhongyuan Wang, Kejun Zhao, Haixun Wang, Xiaofeng Meng, and Ji-Rong Wen, Query Understanding through Knowledge-Based Conceptualization, in *IJCAI*, July 2015.
- Wen Hua, Zhongyuan Wang, Haixun Wang, Kai Zheng, and Xiaofang Zhou, Short Text Understanding Through Lexical-Semantic Analysis, in *International Conference on Data Engineering (ICDE)*, April 2015. **Best Paper Award**
- Fang Wang, Zhongyuan Wang, Senzhang Wang, and Zhoujun Li, Exploiting Description Knowledge for Keyphrase Extraction, in *PRICAI*, December 2014.
- Fang Wang, Zhongyuan Wang, Zhoujun Li, and Ji-Rong Wen, Concept-based Short Text Classification and Ranking, in *ACM International Conference on Information and Knowledge Management (CIKM)*, ACM – Association for Computing Machinery, October 2014.
- Zhongyuan Wang, Haixun Wang, and Zhirui Hu, Head, Modifier, and Constraint Detection in Short Texts, in *International Conference on Data Engineering (ICDE)*, 2014.
- Kai Zeng, Jiacheng Yang, Haixun Wang, Bin Shao, and Zhongyuan Wang, A Distributed Graph Engine for Web Scale RDF Data, in *PVLDB*, August 2013.
- Taesung Lee, Zhongyuan Wang, Haixun Wang, and Seung-won Hwang, Attribute Extraction and Scoring: A Probabilistic Approach, in *International Conference on Data Engineering (ICDE)*, , 2013.
- Peipei Li, Haixun Wang, Kenny Q. Zhu, Zhongyuan Wang, and Xindong Wu, Computing Term Similarity by Large Probabilistic isA Knowledge, in *ACM International Conference on Information and Knowledge Management (CIKM)*, 2013.