中图分类号：TP391
论 文 编 号：10006SY1606405

北京航空航天大學

# 硕 士 学 位 论 文

# 基于知识库和强化学习的代表性话题抽取研究

作 者 姓 名　韩京飞
学 科 专 业　计算机应用技术
指 导 教 师　荣文戈　副教授
培 养 院 系　计算机学院

# Representative Topics Extraction Based on Knowledge Base and Reinforcement Learning

A Dissertation Submitted for the Degree of Master

**Candidate: Jingfei Han**

**Supervisor: Associate Prof. Rong Wenge**

School of Computer Science & Engineering

Beihang University, Beijing, China

中图分类号：**TP391**

论 文 编 号：**10006SY1606405**

硕 士 学 位 论 文

# 基于知识库和强化学习的代表性话题抽取研究

作 者 姓 名　韩京飞　　申请学位级别　工学硕士

指导教师姓名　荣文戈　　职　　称　副教授

学 科 专 业　计算机应用技术　　研 究 方 向　自然语言处理

学 习 时 间 自　2016 年 09 月 01 日　起　　至　2019 年　月　日止

论文提交日期　2019 年　月　日　　论文答辩日期　2019 年　月　日

学位授予单位　北京航空航天大学　　学位授予日期　2019 年　月　日

# 关于学位论文的独创性声明

本人郑重声明：所呈交的论文是本人在指导教师指导下独立进行研究工作所取得的成果，论文中有关资料和数据是实事求是的。尽我所知，除文中已经加以标注和致谢外，本论文不包含其他人已经发表或撰写的研究成果，也不包含本人或他人为获得北京航空航天大学或其它教育机构的学位或学历证书而使用过的材料。与我一同工作的同志对研究所做的任何贡献均已在论文中作出了明确的说明。

若有不实之处，本人愿意承担相关法律责任。

学位论文作者签名：_____ 日期：　　年　　月　　日

# 学位论文使用授权书

本人完全同意北京航空航天大学有权使用本学位论文（包括但不限于其印刷版和电子版），使用方式包括但不限于：保留学位论文，按规定向国家有关部门（机构）送交学位论文，以学术交流为目的赠送和交换学位论文，允许学位论文被查阅、借阅和复印，将学位论文的全部或部分内容编入有关数据库进行检索，采用影印、缩印或其他复制手段保存学位论文。

保密学位论文在解密后的使用授权同上。

学位论文作者签名：_____ 日期：　　年　　月　　日

指导教师签名：_____ 日期：　　年　　月　　日

# 目　　录

# 图　　目

# 表　目

# 第一章　绪论

## 1.1　课题来源与意义

随着大数据时代的到来，互联网数据量呈现指数上升趋势。据国际数据公司（International Data Corp, IDC）的统计和预测，2011 年全球网络数据量已达到 1.8ZB（$1.8 \times 10^6$TB），预计到 2025 年数据总量将会继续增大 50 倍。数据的指数级增长使学者开始关注如何从海量的数据中提取有用知识，这也是大数据分析的关键所在。2012 年 Google 提出知识图谱（Knowledge Graph）概念，并迅速在学术界和工业界普及。基于图结构存储数据的思想普及使最终出现了很多高质量知识库，包括：DBPedia [1, 2]、Yago [3–5]、Wikidata [6]、Microsoft Concept Graph[1]等大型结构化知识库。因此如何利用现有知识库抽取有意义知识受到了广泛关注。

本文旨在利用上下位关系知识库抽取代表性话题。具体来讲就是对于给定学科领域自动抽取最具代表性的 $k$ 个子领域。该问题对于学生、学者、企业、政府等都具有重要意义。对于刚进入某一研究领域的学生而言，一个需要明确的问题是当前的研究热门子领域是什么、各子领域的研究热点问题是什么。本文给出的代表性子领域可以帮助学生明确研究目标，明确研究方向；对于学者而言，可以帮助明确当前的研究热点，并辅助学者寻找潜在优良研究点。比如对于机器学习（Machine learning）与医疗卫生（Health care）领域的交叉学科。可以根据历史数据得到这两个交叉学科的 5 个子领域之间的论文研究趋势，从而判断增长较快的研究点、研究较少的研究点。对于后者，可以分析是由于没有找到合适的方法研究还是未被人关注到，进而确定潜在研究点；对于企业而言，可以推广本文研究，进而构建噪音较少的知识图谱，进而达到更好的应用效果；对于政府而言，研究经费的分配一直是难以确定的问题。本文研究成果可用于找到目前领域内研究重点，进而帮助政府决定研究经费的分配方法。当前学者对文档话题抽取问题进行了大量研究，但是鲜有领域话题抽取的研究。

但是算法抽取的代表性话题并不一定完全符合用户的需求，因此引入用户反馈来改进模型效果是模型的关键一步，同时领域发展情况可以通过用户的反馈体现出来。因此本文希望利用强化学习的方法将用户反馈引入模型，改善模型结果。

[1] https://concept.research.microsoft.com/

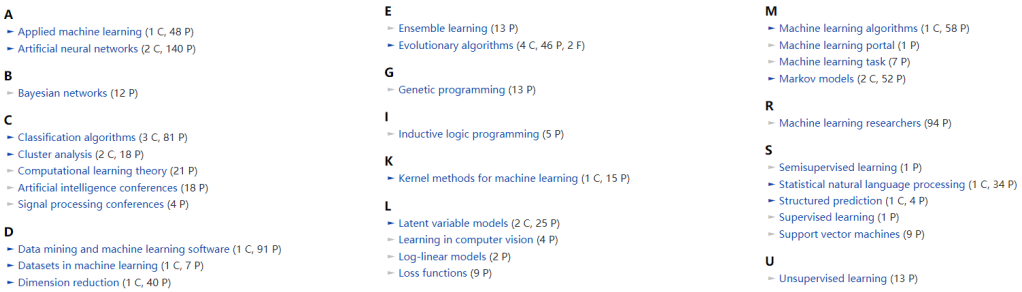


图 1　维基百科中 Machine Learning 的下位词信息

## 1.2　国内外研究现状

本文旨在基于现有知识库抽取领域内代表性话题，并利用用户反馈，应用强化学习方法改进模型结果。下面首先介绍单一上下位关系的可用知识库，然后从问题定义、代表性话题抽取研究、强化学习在 NLP 领域的应用三个角度介绍目前国内外研究现状。

### 1.2.1　上下位关系知识库简介

本文研究问题主要是上下位关系抽取，因此下面将介绍三个单关系知识库：Wiki category [7]，ACM CCS classification tree [8]，以及 Microsoft Field of Study [9]。

Wiki category 是利用维基百科数据得到的 taxonomy 数据集，表示上下位词之间的分类关系。通过数据处理得到一个类似树状结构。之所以说类似树，是以为其中存在少量环路，这是数据存在的问题，因此数据表现为图结构。理论上上下位关系应该是明确的，不应存在环路等情况。经过数据预处理后，可以得到一个由 6,641,759 个实体组成的层次结构。该结构按照维基百科的 category 数据给出上下位关系。比如 Machine Learning 的下位词数据如图1所示[2]。

为说明 taxonomy 结构出存在的异常情况，本文给出一个三层 taxonomy 结构示意图，如图2所示。

图2中红色虚线表示需要说明的情况。

1) **B2→C1**: 该路径目的是说明一个下位词可能存在多个上位词，该情况是合理的，比如一个学科可能在多个领域都有被研究。
2) **B3→B2**: 该路径说明可能兄弟之间存在上下位关系。该情况也是合理的，比如可以认为“人工智能”的两个下位词是“机器学习”和“深度学习”，而“深度学习”也可以认为是“机器学习”的下位词。

[2]https://en.wikipedia.org/wiki/Category:Machine_learning

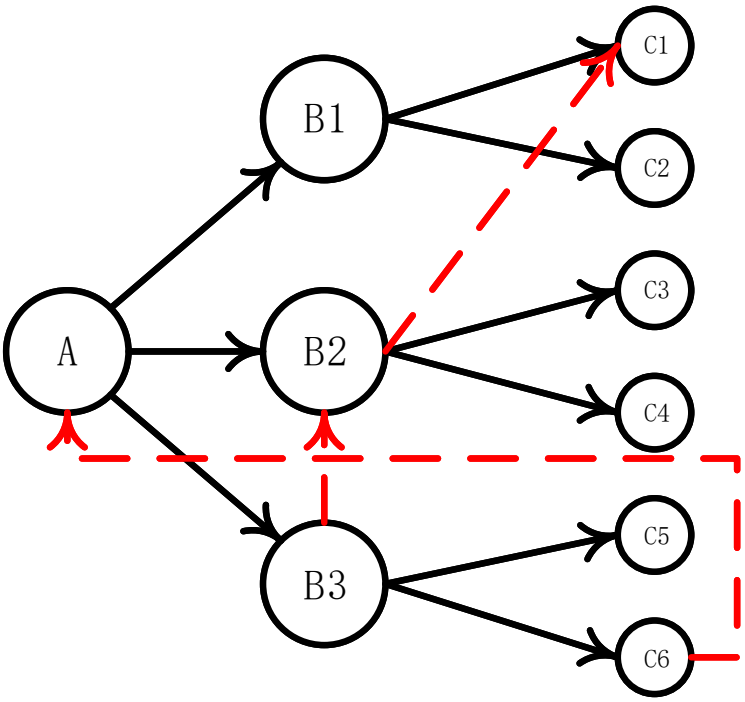


图 2 维基百科 taxonomy 层次化示意图

3) **C6→A**: 该路径表示出现环路，即下位词是其上位词路径上的上位词。这个显然不合理，因为上下位关系本身是一种有向关系，是一种不可逆的关系，而该路径的出现表示上下位关系可逆，因此从逻辑上讲该路径是异常的。维基百科得到的 taxonomy 数据存在这种关系，说明维基百科的数据在上下位关系上是存在噪声的。

经过上述分析可知，正确上下位关系图的基本结构应为有向无环图（Directed Acyclic Graph, DAG）。

ACM CCS （Computing Classification System）分类树 [8] 是一个包含 2126 个节点的层次化实体关系，每个点可以看作一个上位词或下位词。每个非叶子节点都可作为上位词，并且在 CCS 分类树上存在下位词。该系统主要用于学者在 ACM 电子数据库中搜索符合自己需求的文献内容。同时支持将个人文章对应到 CCS 分类下。ACM CCS 存在 14 个顶层分类，包括比如“Hardware”、“Network”、“Software and its engineering”等。层次深度最深达到 4 层。层次化数据呈现树状结构。该知识库优点是对于数据经过人工审查，分类较为仔细，缺点是只针对计算机领域，并不包括其他领域层次化数据，同时 CCS 发布时间是 2012 年，数据量过少，不包括比如“deep learning”等热门领域。因此是一个针对计算机领域的较为准确的数据库。

Microsoft Field of Study 来自微软学术图（Microsoft Academic Graph，MAG）。Field of Study 是微软学术图内置的一个学术上下位词关系的知识库 [9]。包含将近 50000 个

节点，每个节点表示一个学术话题。该知识库是一个 4 层的有向无环图结构，每个下位词可以属于不同的上位词，MAG 对这种情况给出了“可信度”这一指标，表示该下位词以多少“可信度”属于上位词。比如第 3 层节点“Supercomputer”是第 2 层节点“Operating system”的“可信度”是 0.53，是第 2 层节点“Parallel Computing”的“可信度”是 0.47。直觉上讲，超级计算机确实既会被操作系统领域研究，也会被用于并行计算研究。但是都是属于计算机领域的，因此可用 Field of Study 的数据得到“Supercomputer”是“Computer Science”的下位词的“可信度”是 1.0。MAG 的 Field of Study 具有严格的层次结构，它将每个节点（话题）都分成了 L0 至 L3 这 4 个等级。其中 L0 是最高级别的上位词。统计可得各个等级的话题个数，如表1所示。

表 1  MAG 中 Field of Study 数据统计表

| 等级 | 个数 | 示例 |
|---|---|---|
| L0 | 19 | Chemistry, Computer Science, Physics |
| L1 | 290 | Artificial Intelligence, Organic chemistry, Algebra |
| L2 | 1495 | Robotics, Artificial neural network, deep learning |
| L3 | 46851 | Cognitive robotics, Logistic function, Boltzmann machine |

可以看出，L0 层次的话题级别都是学科级别的话题，比如化学、计算机、物理等；L3 层次的话题较为具体，一般为具体的方法等。比如 logistic 函数、波兹曼机等。该知识库实体个数大于 ACM CCS，同时包括针对多种学科的上下位关系，不只局限于计算机领域。

以上介绍了几个常用的大规模开放可下载的知识库。同时针对本文所研究问题，介绍了 3 种存在上下位关系的知识库及优缺点，为了充分利用现有知识库，本文对 Wiki, ACM CCS 和 Microsoft Field of Study 进行融合，得到更大而全的上下位关系知识库。

### 1.2.2　问题定义

Tang 等人 [10] 从大规模社交网络中采样有代表性用户，并对问题进行了形式化定义。其中问题实例如图3所示 [10]。

对于代表性用户采样问题，给出一个社交网络 $G = (V, E)$，其中 $V$ 表示用户集合，表示 $E$ 有向边集合；每个用户定义有 $d$ 个属性；希望在 $V$ 中选出 $k$ 个用户；将属性分为 $t$ 组；质量函数定义为 $Q$；Tang 等人将如何寻找代表性用户集合 $T$ 定义为优化问题，如

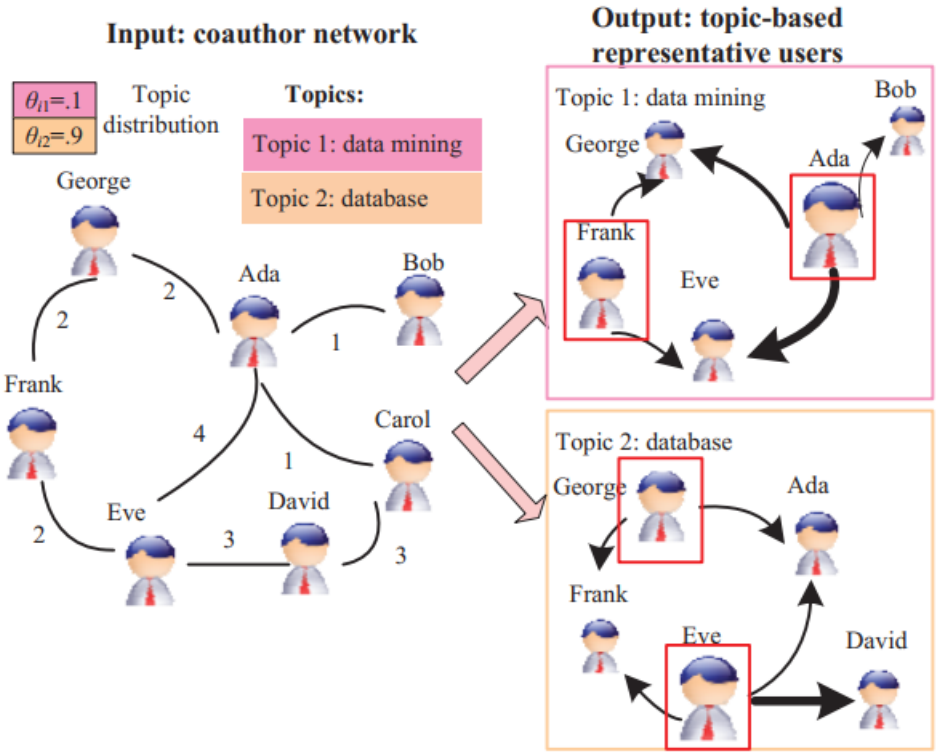


图 3 从合著者网络中采样代表性用户实例

式1.1所示：

$$\arg\max_{T\subseteq V,|T|=k|} Q(G,X,\mathbf{G},T) \tag{1.1}$$

其中 $\mathbf{G}$ 为属性组，$X$ 为属性的分布。$X_i$ 表示用户 $v_i \subseteq V$ 的属性集合，即 $X_i = [x_{ik}]_{k=1,\ldots,d}$。

然后作者通过将问题规约为支撑集问题（Dominating Set Problem）[11]，证明原优化问题为 NP Hard 问题。

### 1.2.3 代表性话题抽取研究

目前主要的代表性话题抽取方法包括主题模型（Topic model）、关键词抽取（Keyphrase extraction）、基于词向量的研究。

对于主题模型，可将话题看作主题模型的主题。目前主题模型已经被大量应用于抽取关键话题。Blei 等人提出 LDA 主题模型 [12]，该模型是一种无监督学习模型，核心思想是每个文档都有多个主题混合组成，而每个单词出现在主题中的概率也不同，主题概率提供了对文档的显式表示。但是 LDA 模型很难给出每个分布表示的具体主题，需要通过大量人工标注。对于该问题，Mei 等人提出将自动标注问题转化为最小化单词分布

间的 KL 散度和最大化标签和主题模型的交互信息的问题。通过试验比较，发现该算法可以高效生成可解释的标签 [13]。Lau 等人针对 LDA 模型提出自动标注话题的方法 [14]。作者利用英文维基百科数据生成标签候选集，然后利用监督学习模型 SVR 得出候选集标签排序，进而实现对话题的自动标注。尽管多位学者对 LDA 话题自动标注进行大量研究并取得一定进展，但是自动标注效果与人工标注存在明显差距。

关键词抽取技术目前主要是两种途径：监督学习和无监督学习。对于监督学习来讲，关键词抽取问题通常被看作分类问题或排序问题。Witten 等人提出 Kea 算法用于从文本中自动抽取关键词。Kea 将关键词抽取问题转化为分类问题，通过使用已知关键词的文档作为训练集，使用训练的模型找出各候选词是否属于当前文档的关键词，进而完成关键词抽取工作 [15]。Jiang 等人认为该问题类似于将候选关键词集合按特征进行排序，排名越靠前说明该关键词对于文档更重要，更可能是文档的关键词，因此作者将关键词抽取问题视为排序问题并使用 learning to rank 的 Ranking SVM 方法完成该任务，关键词抽取效果优于传统 SVM 和 Kea 算法 [16]。

对于无监督学习来讲，Hasan 等人将无监督方法用于解决关键词抽取问题的方法分为四组：基于图的排序法（Graph-Based Ranking）、基于主题的聚类（Topic-Based Clustering）、同时学习（Simultaneous Learning）、语言模型（Language Modeling） [17]。

基于图的排序法最有代表性的是 Kleinberg 等人提出的 HITS 算法 [18] 和 Google 提出的 PageRank 算法 [19]。Mihalcea 等人提出 TextRank 算法用于自动关键词抽取 [20]，将文本看成图，将本文分词，并进行词性标注，用点表示关键词，边表示关键词之间的共现关系构成图。基于 PageRank 算法思想，迭代计算各节点权重，直到收敛。根据迭代结果找出最重要的个单词，即为关键词。

基于主题的聚类法是将关键词分成多个主题组，直觉上每个抽取出来的关键词应该包含文章中的所有主要主题。Liu 等人利用聚类技术找出代表性术语，将每个类的代表性属于作为种子自动抽取关键词。研究表明大部分人工选取的关键词都是名词，因此作者对文档进行词性标注，主要抽取其中的名词和形容词。实验表明准确率在 F1 值上超过 TextRank 9.5% [21]。Hasan 等人指出基于关键词聚类的方法为所有主题分配相同的重要性，实际文档中有些话题并不主要 [17]，因此可能存在缺陷。

同时学习是基于一个假设：重要的词出现在重要的句子中，重要的句子包含重要的词。因此文本摘要和关键词抽取同时学习可能能共同提高效果 [22]。Wan 等人进一步做

出两点假设：1) 关键句与其他关键句相连；2) 关键词与其他关键词相连 [23]。基于上述假设，Wan 等人寄建立三个捕获句子（S）和单词（W）间关系的图：S-S 图、S-W 图和 W-W 图。S-S 图的边权重表示两个句子的相似度；S-W 图的边权重表示词在句子中的重要性；W-W 图的边权重表示两个单词之间的共现次数或者基于知识的相似度。最后通过迭代强化算法为每个句子和单词分配权重，权重高的单词作为关键词 [23]。

语言模型通过两个特征评价关键词：词组性（Phraseness）和信息性（Informativeness）[24]。词组性表示单词序列是否可看作词组；信息性表示哪个单词捕获了文档的中心含义。直觉上讲，有高词组性和信息性的词组更可能是关键词。Tomokiyo 等人通过语言模型分别在前景语料库（Foreground corpus）和背景语料库（Background corpus）上训练得到这两个特征。前景语料库由存在关键词的文档组成，背景语料库是来自 Web 的大量无标注文本数据。通过语言模型得到特征值之后按照两者加和排序候选关键词 [24]。

词向量（Word Embedding）将单词特征进行稠密编码，将特征映射到隐特征空间。Mikolov 等人提出通过预测单词和上下文单词来获取低维特征表示的 Word2Vec 方法 [25]。Word2Vec 的主要包括两种算法：Skip-gram（SG）和连续词袋模型（Continuous Bag of Words, CBOW）。SG 算法是通过当前单词预测上下文单词，CBOW 是通过上下文单词的词袋模型预测目标单词。训练时为了更加高效可以使用两种算法：层次化 Softmax（Hierarchical Softmax）和负采样（Negative Sampling）。因为本文研究是基于词向量相似度的代表性话题抽取，因此下面重点介绍 Skip-gram 模型的工作流程。

Skip-gram 模型通过当前单词预测周围单词出现的概率。如图4所示 [25]。

Skip-gram 通过单隐层神经网络完成预测周围单词出现概率的任务。但这实际是一个“伪任务”[?]。即真正想要的并不会真正使用模型去完成上述预测任务，而是为了得到隐层权重，因此实际“伪任务”得到的权重就是通过 Skip-gram 模型得到的词向量。

输出的概率表示每个单词更有可能在输入单词附近的概率。比如给出输入单词“Soviet”，周围的单词是“Union”或者“Russia”的概率较大，而是“西瓜”或者“袋鼠”的概率就很小 [?]。而在输入单词附近由“窗口大小”衡量。比如存在句子“The quick brown fox jumps over the lazy dog.”，假设窗口大小 $c = 2$。表示在中心词前后各选择两个单词作为环境词。最终得到训练样本 < 输入，输出 > 为：<the, quick>, <the, brown>, <quick, the>, <quick, brown>, <quick, fox>, <brown, the>, <brown, quick>, <brown, fox>, <brown, jumps>, <fox, quick>, <fox, brown>……后面模式相同。通过训练，即可通过训

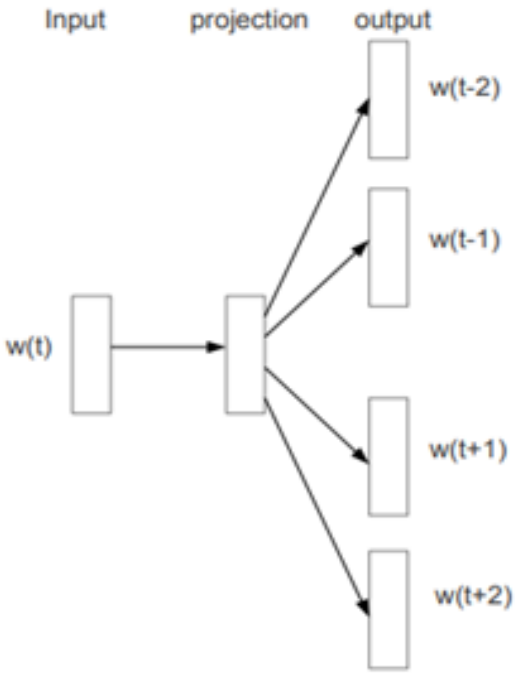


图 4　Skip-gram 模型结构

练好的模型预测当前单词周围的词汇出现的概率。

更一般地，给出句子序列 $S =< w_1, w_2, ..., w_T >$，其中 $T$ 为序列长度。利用最大似然估计可得，Skip-gram 的目标是最大化式：

$$J(\theta) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0}\log p(w_{t+j}|w_t) \tag{1.2}$$

其中 $c$ 是训练窗口大小，$c$ 越大训练数据越多，准确率也越高，但是耗费时间也越大。式1.2中的 $p(w_{t+j}|w_t)$ 使用了 softmax 函数表示，如式1.3所示。

$$p(w_O|w_I) = \frac{\exp({v'_{wO}}^T v_{wI})}{\sum_{w=1}^{W}\exp({v'_w}^T v_{wI})} \tag{1.3}$$

其中 $v_w$ 和 $v'_w$ 分别表示单词 $w$ 的输入、输出向量表示。$W$ 表示词表大小。

实际该模型可以展示为一个三层的神经网络。其中隐含层为线性激活单元，输出层为 Softmax 分类器。具体结构如图5所示 [? ]。

输入层到隐含层的参数 $W$ 维度为 $|V|\times|H|$，即词表大小乘以隐含层大小。$w(i,:)$ 可以看作第 $i$ 个单词的词向量。如果两个不同的单词有相似的上下文，即出现在其周围的单词较为相似，则 Skip-gram 模型将输出相似的结果。因此如果两个单词上下文相似，则模型倾向于学习相似的词向量。

因此可知，词意相似的单词词向量也比较相似，而可以认为下位词的词向量应与上

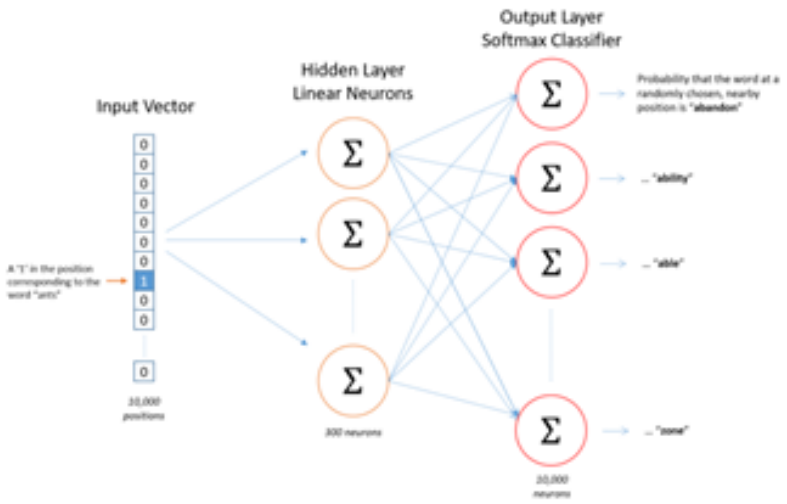


图 5 Skip-gram 网络结构示意图

位词词向量相似，进而实现代表性话题抽取。尽管存在 Glove 等更优秀的单词分布式表示方法[26]，但是由于效率和易用性等原因，还是考虑使用 Skip-gram。

### 1.2.4 强化学习在 NLP 领域的应用

近年来，强化学习（Reinforcement Learning）与深度学习结合，开始成为研究热点问题[27]。因此强化学习也在各领域大放异彩。2015 年 10 月 AlphaGo 击败欧洲围棋冠军，2016 年 3 月击败 18 次获得世界围棋比赛冠军的李世石，2017 年 5 月击败围棋世界第一柯洁。并且提出完全自我训练的 AlphaGo[28]。2017 年 12 月，Deep Mind 团队公开 AlphaZero[29]，同时在多种棋类上做出大量提升。这是强化学习最引人注目的应用之一[30]。下面本文先简单介绍强化学习的基本知识和核心概念，然后介绍近年来强化学习在 NLP 领域的应用。

强化学习是 Agent 与环境（Environment）交互的过程，通过学习最优策略（Policy）和试错（Trail and error）解决序列决策问题（Sequential decision making problem）。强化学习问题在于学习做什么才能最大化所得奖赏（Reward），即如何将当前状态（State）映射为动作（Action）。本质上强化学习问题是一个闭环问题（Closed-loop），因为学习系统的行为会影响下一次的输入[31]。强化学习过程示意图如图6所示[3]。

图6展示了强化学习的交互过程，其中，在第 $t$ 步，agent 执行动作 $A_t$，收到观测 $O_t$ 等价于上文所述的 State$S_t$），收到标量奖赏 $R_t$。环境收到动作 $A_t$，发出观测 $O_{t+1}$（等价于 $S_{t+1}$），发出标量奖赏 $R_{t+1}$。$t$ 在每次交互完成后递增。

[3] goo.gl/UqaxlO

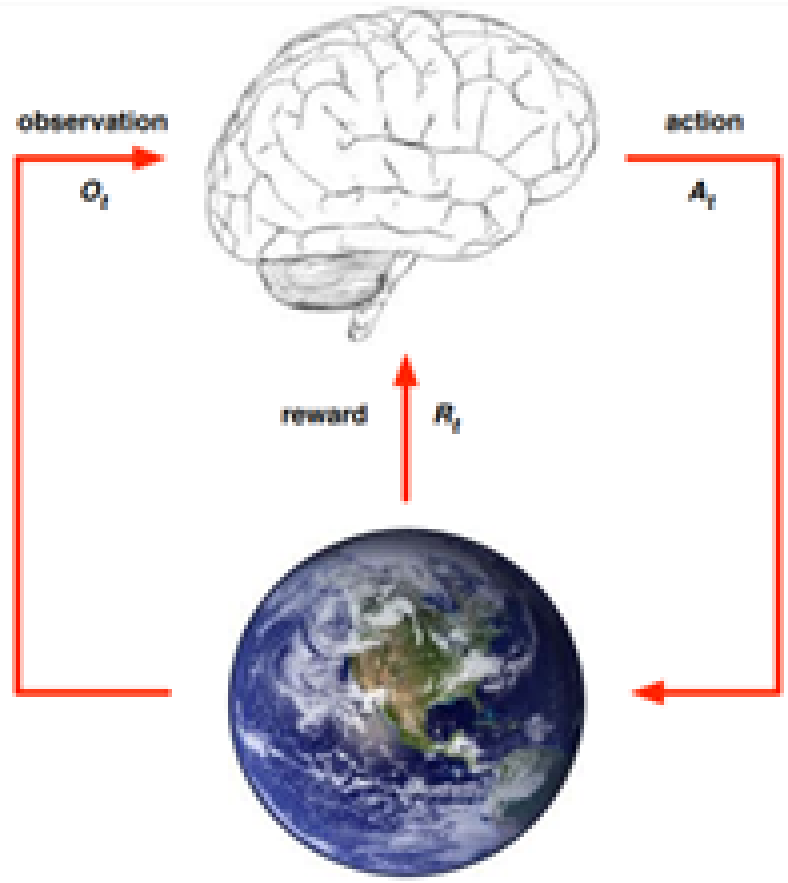


图 6　强化学习 Agent 和 Environment 交互过程

定义当前时刻 $t$ 的累计奖赏为：

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \tag{1.4}$$

其中 $\gamma \in (0, 1]$ 为折扣因子（discount factor），$r_t$ 表示 $t$ 时刻的即时奖赏。

价值函数（Value Function）用于衡量当前状态的好坏程度，可用累计奖赏的期望表示。即：

$$v_\pi(s) = E[R_t | s_t = s] \tag{1.5}$$

其中 $v_\pi(s)$ 为在状态 $s$ 下使用策略 $\pi$ 的期望累积奖赏值。将 $v_\pi(s)$ 分解为 Bellman 等式可得：

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')] \tag{1.6}$$

最优状态值可定义为:

$$v_*(s) = \max_\pi v_\pi(s) = \max_a q_{\pi*}(s, a) \tag{1.7}$$

若强化学习问题满足 Markov 性质，即下一步状态只依赖于当前状态和动作，与过去无关，则可以将问题看作 Markov 决策过程（Markov Decesion Process，MDP），用五元组 $(S, A, P, R, \gamma)$ 表示 [32]。对于有模型学习（Model-based learning）问题，机器已对环境进行建模，状态、动作、状态转移概率、转移奖赏函数均为已知，则可考虑使用动态规划方法：策略评估（Policy evaluation）计算策略的价值函数，策略迭代（Policy iteration）或值迭代（Value iteration）找到最优策略。对于免模型学习（Model-free learning）问题，环境转移概率、奖赏函数、甚至不知道环境中一共有多少状态。此时可以使用蒙特卡洛方法，根据执行该策略 $T$ 步得到的采样轨迹 $< s_0, a_0, r_1, s_1, a_1, ..., s_{T-1}, a_{T-1}, r_T, s_T >$ 可得到状态-动作值函数的估计。另一种解决方法是时序差分学习（Temporal Difference，TD），同时利用采样轨迹和动态规划结构，得到值函数或状态-动作值函数的估计。SARSA 算法 [33] 和 Q-Learning 算法 [34] 是经典的 TD learning 算法 [35]。

之前提到的值函数或状态-动作值函数均被以表格形式存储，但是当状态空间很大或者连续时，内存需求会太大甚至无法存储，因此需要使用近似值函数（Function Approximation）估计值函数。这种近似函数可以是线性函数或非线性函数，DQN 则通过使用 CNN 实现端到端的强化学习 [27, 36]，实现直接从像素输入学习游戏打法并超过顶尖人类玩家水平。

除了 TD learning 和 Q-learning 等基于值的方法（Value-based methods），还有基于策略的方法（Policy-based methods）。基于策略的方法目标是直接优化策略 $\pi(a|s;\theta)$，其中 $\theta$ 是近似函数的参数，并通过梯度上升（Gradient ascent）方法优化 $E[R_t]$。Williams 等人提出的 REINFORCE 算法是一种经典策略梯度（Policy gradient）算法，在方向 $\nabla_\theta \log \pi(a_t|s_t;\theta)$ 上更新参数 $\theta$。

强化学习在多个 NLP 子领域开始被应用，包括对话系统（Dialogue system）、机器翻译（Machine translation）、文本生成（Text generation） [32, 37]。在对话系统领域，Li 等人基于监督学习和强化学习提出端到端的神经对话系统。该框架包括用户仿真器和神经对话系统。作者利用强化学习端到端的训练系统，将对话策略看作 DQN，进而降低其他模块产生的噪声 [38]。在机器翻译领域，He 等人提出对偶学习处理机器翻译领域数据不足的问题。机器翻译从语言 A 到语言 B 和对偶任务，从语言 B 到语言 A，可以帮助提高两个翻译模型性能。作者使用策略梯度方法，使用语言模型似然度作为奖励函数。实验表明该模型只需要较少数据即可实现优越性能 [39]。文本生成是多种 NLP 任

务的基础，比如对话生成、机器翻译、自动摘要等。Bahdanau 等人使用强化学习算法 actor-critic 实现序列预测，使用 critic 网络预测单词的价值，即 actor 网络定义的序列预测策略的期望累计价值 [40]。除此之外，强化学习开始被用于微调（Fine-tuning）模型结果。Zhang 等人使用 RNN decoder 和 attention 机制对病人病症生成治疗药物，保证药物之间不出现药物冲突。使用策略梯度方法基于规则去除仍存在冲突的药物组合，实验表明强化学习可以实现机制冲突药物组合的产生 [41]。

## 1.3　论文研究内容

在历史文献当中，源词和目标词分别被称为模型的输入和输出。源词通常可以用分布式表示（Distributed Representation）来表示，称为输入词嵌入（Word Embedding），通常在大规模语料上使用连续词袋模型（CBoW）、跳跃单词模型（Skipgram）或者 Fasttext 模型来训练。而这几种模型均起源于语言建模任务，并且考虑到实际运算的效率而去除了部分冗余结构。另外，输出端的词表通常表示为单词索引（Indexing）或 $1-K$ 编码，并且可以与 softmax 概率函数直接关联。

在语言模型研究领域中，大词表问题是目前理论应用到实际过程中必须要克服的问题，我们当然可以通过配置高性能服务器来缓解该计算瓶颈。一旦应用到较大规模的数据集上，即使是目前最好的中央处理器（CPU）或者通用计算图形处理器（GPGPU），仍然需要数周时间才能训练完善。因此，在保证原有模型的准确率和精度的前提下，如何提高模型的训练速度是本文主要讨论和研究的内容。为此我们考察了两个主要的研究目标：上下文信息建模效率和精度对比、大词表问题的优化和研究。

针对大词表问题的优化，目前主要采用的方案有以下几种：一种是采用子词（Subword-level）或者字符级别的词（Character-level）来直接缩小词表大小；一种是通过采样技术（Sampling-based Approximation）来减少必要的训练时间；另一种是通过基于分类的多元分类（class-based Hierarchical Softmax，cHSM）和采用基于树模型的多层二元分类模型（tree-based Hierarchical Softmax，tHSM）来加速模型。

本论文考虑了层次概率模型所存在的一些问题，并提出相应的解决策略。首先，我们提出了一个在分层结构上建模参数的单词编码方案，推导出紧凑的代价函数及其梯度。同时考虑到类或树上的单词分布对其性能有很大的影响，我们运用文本的统计，句法和语义知识来初始化其参差结构，以达到稳定的计算精度。同时在推理过程中，我们

考虑了两种不同的推理情况：句子打分和文本排序，并提出了对应的优化策略。

## 1.4 论文的组织结构

第一章：“绪论”，主要介绍了本论文的研究背景和意义，另外简要说明了语言模型的发展历史以及本文的主要工作，并对本文的组织架构进行了说明。

第二章：“相关技术介绍”，对历史上的各个学术分支在语言模型的任务上的相关工作进行了介绍。

第三章：“树状层次概率模型”，介绍了基于二叉树的层次概率模型，并与传统树状模型做了理论层面的比较。同时还研究了在推理测试阶段，二叉树层次概率模型应用的贪心策略，以保证实际测试结果性能和效率。

第四章：“类别层次概率模型”，介绍基于分类的层次概率模型，并分析了词表非均匀划分所产生的后果，进而探讨了类别不均匀问题所带来的影响以及相关解决策略。最后探讨了在测试阶段，语言模型的任务需求和分类层次概率模型相应的解决算法。

第五章：“语言建模实验及结果分析”，实证研究了本论文提出的层次概率模型的实际效果，并和其他算法在各个指标维度上进行了比较和分析。

最后结论部分，总结了全论文的贡献和工作，并提出了未来的工作方向，同时撰写了结束语。

# 参考文献

[1] Auer S, Bizer C, Kobilarov G, et al. Dbpedia: A nucleus for a web of open data[G]. The semantic web[C]. [S.l.]: Springer, 2007: 722–735.

[2] Zaveri A, Kontokostas D, Sherif M A, et al. User-driven quality evaluation of dbpedia[A]. Proceedings of the 9th International Conference on Semantic Systems[C]. 2013: 97–104.

[3] Suchanek F M, Kasneci G, Weikum G. Yago: a core of semantic knowledge[A]. Proceedings of the 16th international conference on World Wide Web[C]. 2007: 697–706.

[4] Hoffart J, Suchanek F M, Berberich K, et al. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia[J]. Artificial Intelligence, 2013, 194: 28–61.

[5] Mahdisoltani F, Biega J, Suchanek F M. Yago3: A knowledge base from multilingual wikipedias[A]. CIDR[C]. 2013.

[6] Vrandečić D, Krötzsch M. Wikidata: a free collaborative knowledgebase[J]. Communications of the ACM, 2014, 57(10): 78–85.

[7] Ponzetto S P, Strube M. WikiTaxonomy: A Large Scale Knowledge Resource.[A]. ECAI: Vol 178[C]. 2008: 751–752.

[8] Coulter N, French J, Glinert E, et al. Computing classification system 1998: current status and future maintenance. Report of the CCS update committee[J]. Computing Reviews, 1998, 39(1): 1–62.

[9] Sinha A, Shen Z, Song Y, et al. An overview of microsoft academic service (mas) and applications[A]. Proceedings of the 24th international conference on world wide web[C]. 2015: 243–246.

[10] Tang J, Zhang C, Cai K, et al. Sampling Representative Users from Large Social Networks.[A]. AAAI[C]. 2015: 304–310.

[11] Garey M R. A Guide to the Theory of NP-Completeness[J]. Computers and intractability, 1979.

[12] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan) : 993 – 1022.

[13] Mei Q, Shen X, Zhai C. Automatic labeling of multinomial topic models[A]. Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining[C]. 2007 : 490 – 499.

[14] Lau J H, Grieser K, Newman D, et al. Automatic labelling of topic models[A]. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1[C]. 2011 : 1536 – 1545.

[15] Witten I H, Paynter G W, Frank E, et al. KEA: Practical Automated Keyphrase Extraction[G]. Design and Usability of Digital Libraries: Case Studies in the Asia Pacific[C]. [S.l.] : IGI Global, 2005 : 129 – 152.

[16] Jiang X, Hu Y, Li H. A ranking approach to keyphrase extraction[A]. Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval[C]. 2009 : 756 – 757.

[17] Hasan K S, Ng V. Automatic keyphrase extraction: A survey of the state of the art[A]. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) : Vol 1[C]. 2014 : 1262 – 1273.

[18] Kleinberg J M. Authoritative sources in a hyperlinked environment[J]. Journal of the ACM (JACM), 1999, 46(5) : 604 – 632.

[19] Brin S, Page L. The anatomy of a large-scale hypertextual web search engine[J]. Computer networks and ISDN systems, 1998, 30(1-7) : 107 – 117.

[20] Mihalcea R, Tarau P. Textrank: Bringing order into text[A]. Proceedings of the 2004 conference on empirical methods in natural language processing[C]. 2004.

[21] Liu Z, Li P, Zheng Y, et al. Clustering to find exemplar terms for keyphrase extraction[A]. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1[C]. 2009 : 257 – 266.

[22] Zha H. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering[A]. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval[C]. 2002 : 113 – 120.

[23] Wan X, Yang J, Xiao J. Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction[A]. Proceedings of the 45th annual meeting of the association of computational linguistics[C]. 2007 : 552 – 559.

[24] Tomokiyo T, Hurst M. A language model approach to keyphrase extraction[A]. Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18[C]. 2003 : 33 – 40.

[25] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[A]. Advances in neural information processing systems[C]. 2013 : 3111 – 3119.

[26] Pennington J, Socher R, Manning C. Glove: Global vectors for word representation[A]. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)[C]. 2014 : 1532 – 1543.

[27] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.

[28] Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. nature, 2016, 529(7587) : 484.

[29] Silver D, Hubert T, Schrittwieser J, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm[J]. arXiv preprint arXiv:1712.01815, 2017.

[30] Silver D, Hassabis D. AlphaGo: Mastering the ancient game of Go with Machine Learning[J]. Research Blog, 2016.

[31] Sutton R S, Barto A G, Bach F, et al. Reinforcement learning: An introduction[M]. [S.l.] : MIT press, 1998.

[32] Li Y. Deep reinforcement learning: An overview[J]. arXiv preprint arXiv:1701.07274, 2017.

[33] Rummery G A, Niranjan M. On-line Q-learning using connectionist systems : Vol 37[M]. [S.l.] : University of Cambridge, Department of Engineering Cambridge, England, 1994.

[34] Watkins C J, Dayan P. Q-learning[J]. Machine learning, 1992, 8(3-4) : 279 – 292.

[35] Arulkumaran K, Deisenroth M P, Brundage M, et al. A brief survey of deep reinforcement learning[J]. arXiv preprint arXiv:1708.05866, 2017.

[36] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540) : 529.

[37] Williams R J. Simple statistical gradient-following algorithms for connectionist reinforcement learning[J]. Machine learning, 1992, 8(3-4) : 229 – 256.

[38] Li X, Chen Y-N, Li L, et al. End-to-end task-completion neural dialogue systems[J]. arXiv preprint arXiv:1703.01008, 2017.

[39] He D, Xia Y, Qin T, et al. Dual learning for machine translation[A]. Advances in Neural Information Processing Systems[C]. 2016 : 820 – 828.

[40] Bahdanau D, Brakel P, Xu K, et al. An actor-critic algorithm for sequence prediction[J]. arXiv preprint arXiv:1607.07086, 2016.

[41] Zhang Y, Chen R, Tang J, et al. Leap: learning to prescribe effective and safe treatment combinations for multimorbidity[A]. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining[C]. 2017 : 1315 – 1324.